Information Technology Solutions

Managing LANs

DATAPRO

Managing LANs

DATAPRO
Information Services
Group

600 Delran Parkway
P.O. Box 1066
Delran, New Jersey 08075
USA
Tel: (800) 328-2776
Tel: (609) 764-0100
Fax: (609) 764-2814

Australia:
Datapro International
Level 5
3 Spring Street
Sydney NSW 2000
Australia
Tel: (61) 2 2522311
Fax: (61) 2 2477992

Canada:
Datapro International
270 Yorkland Boulevard
North York, Ontario
M2J 1R8 Canada
Tel: (800) 668-9308
Tel: (416) 496-3131
Fax: (416) 496-3160

Frankfurt:
Datapro International
Liebigstrasse 19
D-6000 Frankfurt 1
Germany
Tel: (49) 69 714070
Fax: (49) 69 71407146

Hong Kong:
Datapro International
16/F Unit C
Cindic Tower
128 Gloucester Road
Wanchai, Hong Kong
Tel: (852) 8022040
Fax: (852) 8360697

Japan:
Nikkei Business Publications
1-14-6, Uchikanda
Chiyoda-ku, Tokyo 101
Japan
Tel: (81) 03-233-8081
Fax: (81) 03-233-3048

Paris:
Datapro International
128 rue du Faubourg Saint-Honore
F-75008 Paris
France
Tel: (33) 1 42890381
Fax: (33) 1 42890400

Singapore:
Datapro International
Unit 05-03 Dapenso Building
158 Cecil Street
Singapore 0106
Tel: (65) 2225091
Fax: (65) 2213321

South America:
MIPS
Rua Camargo 12
Ossio-Sao Paulo-SP
Brazil
Tel: (55) 11 813 7492
Fax: (55) 11 813 7476
814 4363

United Kingdom:
Datapro International
McGraw-Hill House
Shoppenhangers Road
Maidenhead
Berkshire SL6 2QL Great Britain
Tel: (44) 628 773277
Fax: (44) 628 773628

Contents

Volume 1
February 1992

Managing LANs

(Formerly Datapro Management of Microcomputer Systems)

Report Number	Report
4827	Using SNA to Link LAN to Host
4828	PC Links to UNIX Database Servers
4830	LAN Applications
4835	LAN Internetworking
4837	Planning and Implementing a Local Area Network Internet
4840	Implementing CD-ROM Applications on a LAN
4850	Migrating LAN Workstations From DOS to OS/2
4855	The Client/Server Iterative Rapid Development Methodology
4860	Multitasking on the 386
4862	Programming on the 386
4864	Iconic Programming

G

H

I

J

K

L

M

N

O

P

R

S

T

U

V

W

Y

Getting Started

Put *Managing LANs* to Work for You
Whether you're incorporating new technologies into an existing LAN, or assessing the issues and costs surrounding the implementation of an entirely new one, Datapro's *Managing LANs* can help you in your strategic computing decision-making process. *Managing LANs* can help you anticipate changes implied by emerging technologies. It can also help you make sure all relevant business trends are taken into account as part of your planning process, and ensure that your LAN decisions support your organization's long-term computing and communications goals.

Managing LANs offers a single, comprehensive management information resource with the breadth and depth to help managers take advantage of all the developments and management techniques to meet business goals. *Managing LANs* provides essential information on planning, designing, implementing, and managing a local area network computing environment.

In this "Getting Started" section, you will find everything you need to use *Managing LANs* to your full advantage.

What's Included?
Your *Managing LANs* information service consists of a two-volume reference set. Every month you will receive a monthly issue that will keep you continually updated and informed on the current and emerging technologies, LAN technologies and business issues, proven LAN management techniques and concepts, and events affecting the industry. As a *Managing LANs* customer, you also receive free, unlimited access to the Datapro Help Desk, a telephone inquiry service to answer both general and specific questions, including assistance in using *Managing LANs*.

Keeping You Informed
Managing LANs is a guide to issues, trends, and proven techniques for managing a LAN computing environment. It provides the information you need to plan and design an efficient and cost-effective local area network. It also gives you the technology reviews on important LAN components to help with your selection process.

There are many ways you can use and benefit from *Managing LANs*. Technology issues and trends help you evaluate the most current technology trends and provide guidelines for effective integration of products supporting those technologies into a fully functioning LAN computing environment. Business issues help managers ensure that their LAN decisions support their organization's long-term computing and communications goals. Comprehensive and integrated planning and design guidelines enable you to plan more strategically and make well-informed decisions to minimize risk and maximize flexibility. Architecture and standards reports help you plan strategically for the incorporation of a LAN computing environment within your organization.

By using *Managing LANs* as your LAN computing guide you can make decisions that ensure the success of your organization's LAN strategies.

Find It Fast
Managing LANs is designed to help you locate information quickly. The reports in each monthly issue are arranged by subject and identified by an easy-to-use report numbering system that enables you to locate them easily. The binders provided with your subscription contain clearly labeled tabs behind which these reports can be filed. Once you read the monthly issue, we recommend filing the reports in the binders to help you locate important information in the future. The Table of Contents that appears in each issue includes section heads that correspond to the labeled tabs and identify specific subject areas.

Index
The Index (Tab 100) supports the Contents. It provides the means to quickly locate information on very specific topics. Issues, concepts, technologies, standards, and architectures are cross-referenced to help you quickly find what you seek.

Help Desk
Datapro offers free access to the Help Desk to all registered customers. The Help Desk is a quick reference tool to help you with:
- Information on new products
- Detailed pricing, when available
- Information typically found in annual reports
- Product searches
- Using Datapro information services

To register for the Help Desk please complete the registration form located behind the Help Desk & Customer Support tab.

Report Numbering System
Managing LANs reports are identified by a 3- or 4-digit number.

The 3-digit report numbers (100, 200, 300, and 400) refer to the Index, Getting Started, Help Desk & Customer Support, and Other Datapro Information Services sections, respectively.

The 4-digit report numbers are divided into two parts. The first two digits refer to the section, or tab, within the information service; the second two digits are used to serially file and locate the reports within the section.

For example, in *Managing LANs*, section 42 refers to planning, and report number 05 is Planning a LAN Implementation.

Datapro Report Types
Managing LANs includes various types of reports, each designed to help you quickly find the information you need.

Issues & Trends
These reports bring LAN managers up to date quickly on what's happening in the industry in order to take advantage of the technology and business changes. They can help ensure a more effective use of local area networking throughout the organization.

Management Reports
Management reports provide guidelines to facilitate the planning, design, selection, implementation, and operations associated with managing a local area network computing environment. They offer practical advice and methodologies from industry experts to help you minimize risk and improve the effectiveness and efficiency of your LAN.

Overviews
The overviews are a comprehensive summary of particular LAN components. The overview reports provide technology basics and selection criteria to enable you to make well-informed and strategic buying decisions.

Architectures & Standards
These reports keep managers up to date on the major LAN standards and architectures. This information is critical to ensure that the LAN environment is efficiently configured and effectively integrated into the company-wide network. ■

Help Desk & Customer Support

Datapro believes that it is critical to be close to you, the customer, and to proactively serve your needs. To ensure that you receive the most from your information service, Datapro offers the Datapro Help Desk and a dedicated customer service department, as well as a low-cost option to help you maintain your Datapro products without having to file updates. We also conduct periodic customer surveys and encourage you to give us feedback on an ongoing basis.

Help Desk: Someone to Turn to When You Need an Answer

As a Datapro information service customer, you receive free, unlimited access to the Datapro Help Desk, a telephone reference service unique to the industry. The Help Desk consultants have an unrivaled database of industry information at their disposal. They can provide:

- Late-breaking news on new products and product enhancements
- Detailed pricing, when available
- Vendor financial statements
- Product searches
- Assistance with interpreting technical specifications
- Help using your Datapro service
- Information about other Datapro services

While the Help Desk can handle a broad range of requests, information needs requiring significant original research are handled on a special contract basis through Datapro's On-Call Consulting Department (1-800-328-2776). For example, you might need a search of archived reports on in-depth analysis of products and features not covered in a published information service.

To register for the Help Desk, you need to complete the Help Desk registration form. There's a form on page 310 in this section in case you haven't registered yet. Simply fax the completed form to Datapro at 609-764-0451, or drop it in the mail. Then you can begin using the Help Desk right away.

Your Customer Service Rep: Someone You Should Know

Datapro's Customer Service Department is fully staffed to ensure that you are completely satisfied with your Datapro information service. If you ever have a question or problem, your Customer Service Representative is just a toll-free call away: 1-800-328-2776.

Call your Customer Service Rep for:

- A change of address for shipping or billing. See the postage-paid Change of Address form on page 320 of this section.
- Help in resolving a billing problem.
- Replacing lost or damaged material. For a nominal charge, Datapro will replace missing pages, lost issues, and damaged binders.
- Information about our Maintenance Service.
- Suggestions or complaints about your information service. (Also see the Customer Feedback form on page 330 of this section.)

Tell Us What You Think

You can help us serve you better by telling us what you think about your Datapro information service. We want your feedback--both positive and negative--and encourage you to let us know how we're doing.

To make it easy for you, we've included a postage-paid Customer Feedback Form on page 330 of this section. Simply fill it out and either fax it to us or drop it in the mail. Or, if you prefer, just call your Customer Service Rep or the Help Desk.

However you do it, we'd like to hear from you.

Say Goodbye to Filing: Datapro Maintenance Service

Filing monthly issues keeps your information service well organized and up-to-date, but some customers can't spare the time filing requires. Datapro now offers a low-cost Maintenance Service that, in effect, does your filing for you. Here is how it works:

- **Every month** you receive your regular update. You read the reports, distribute them to colleagues, file reports in project folders. Make full use of them. Don't worry about filing them because you don't have to file anything.
- **Every quarter** Datapro sends you the complete, up-to-date contents of your information service to replace the full contents of your binders. You receive all current reports and indexes. You even receive new divider tabs.

With Maintenance Service, your information service will remain current, and you'll never have to file another monthly update.

Call your Customer Service Rep at 1-800-328-2776 to find out more about Datapro Maintenance Service.

Help Desk Registration Form

Datapro Help Desk: Meeting your unique information needs. When you order any Datapro Information Service, you'll get **free, unlimited access** to the Datapro Help Desk, a telephone reference service unique to the industry. The Datapro Help Desk consultants have an unrivalled database of industry information at their disposal. Whether you need late-breaking product news, assistance with interpreting technical specifications, help using your Datapro service, vendor financial statements or other information, you can expect prompt and reliable service from Datapro. To use the Help Desk, simply indicate **up to two additional authorized users** and return this form via fax (609-764-2813) or use the self-mailer on the back of this form. Use of this service will be limited to registered customers.

NAME ______________________
TITLE ______________________
COMPANY ______________________
ADDRESS ______________________
CITY __________ **STATE** __________ **ZIP** __________
TELEPHONE NUMBER (______) ______________
FAX NUMBER (______) ______________
DATAPRO PRODUCTS RECEIVED: ______________________
____
____
ACCOUNT NUMBER: ______________________
AUTHORIZED SIGNATURE: ______________________
NAMES OF TWO ADDITIONAL AUTHORIZED USERS OF THE HELP DESK REFERENCE SERVICE.
Name __________ **Name** __________
Title __________ **Title** __________

1. BUSINESS/INDUSTRY

The principal business of your firm at this location.
Check only one.

- ☐ 1. Agriculture/Mining/Construction
- ☐ 2. Manufacturing - Computers, Communication
- ☐ 3. Other Manufacturing
- ☐ 4. Communication/Carriers/ Interconnects
- ☐ 5. Transportation/Utilities
- ☐ 6. VARs/Distributors, Computers and Communication
- ☐ 7. Wholesale/Retail Trade
- ☐ 8. Finance/Banking
- ☐ 9. Insurance/Real Estate
- ☐ 10. Professional Services
- ☐ 11. Business Services
- ☐ 12. Consulting
- ☐ 13. Other Services (specify) __________
- ☐ 14. Education/Health Care
- ☐ 15. Government (Federal, Military)
- ☐ 16. Government (State, Local)
- ☐ 17. Other (Please specify) __________

2. JOB TITLE

Check only one.

- ☐ 1. Chairperson, President, Partner
- ☐ 2. Vice President
- ☐ 3. Director/Department Head
- ☐ 4. Manager/Supervisor
- ☐ 5. Professional Staff
- ☐ 6. Librarian
- ☐ 7. Technical staff
- ☐ 8. Engineer
- ☐ 9. Consultant
- ☐ 10. Systems Integrator
- ☐ 11. Other (Please specify) __________

3. YOUR DEPARTMENT

Check only one.

- ☐ 1. Corporate Management
- ☐ 2. MIS/DP
- ☐ 3. Data/Telecommunications
- ☐ 4. Network/Systems Mngmnt.
- ☐ 5. Office Automation
- ☐ 6. Manufacturing
- ☐ 7. Operations/Administration Services
- ☐ 8. Library/Information Center
- ☐ 9. Sales
- ☐ 10. Marketing
- ☐ 11. Engineering
- ☐ 12. Finance/Accounting
- ☐ 13. Purchasing
- ☐ 14. Technical Support
- ☐ 15. Other (Please specify) __________

4. NUMBER OF EMPLOYEES

Check one in each column.

	Location	Company
Under 50	☐	☐
50 - 99	☐	☐
100 - 499	☐	☐
500 - 999	☐	☐
1,000 - 4,999	☐	☐
5,000 or more	☐	☐

5. ESTIMATED REVENUES FOR YOUR COMPANY

Check only one.

- ☐ Under $10 million
- ☐ $10 to $24.9 million
- ☐ $25 to $99.9 million
- ☐ $100 to $249.9 million
- ☐ $250 to $499.9 million
- ☐ $500 to $999.9 million
- ☐ $1 billion or more

6. PURCHASE INFLUENCE

Which products do you recommend, specify, or purchase?
Which do you plan to purchase in the next year?
Check all that apply.

Computers/Peripherals

Recommend	Plan to Purchase
☐ Mainframes	☐
☐ Minicomputers	☐
☐ Microcomputers	☐
☐ Terminals	☐
☐ Printers	☐
☐ Factory Floor Equipment	☐
☐ Workstations	☐

Software

Recommend	Plan to Purchase
☐ Systems software	☐
☐ Applications software	☐
☐ Communications software	☐
☐ Network operating systems	☐
☐ Network management	☐

Networking Equipment

Recommend	Plan to Purchase
☐ Local area networks	☐
☐ Modems	☐
☐ T-1 equipment	☐
☐ PBX	☐
☐ Packet switches	☐
☐ Network management switches	☐
☐ Central office switches	☐
☐ Fiber optic equipment	☐

Disaster Recovery/Information Security

Recommend	Plan to Purchase
☐ Access Control Software	☐
☐ Disaster Recovery	☐
☐ Facilities Planning	☐
☐ Physical security control	☐

Other

☐ __________	☐
☐ __________	☐

7. OPERATING SYSTEM(S) USED

Check all that apply.

- ☐ MS-DOS
- ☐ OS/2
- ☐ UNIX
- ☐ Apple OS
- ☐ Macintosh
- ☐ DG AOS/VS
- ☐ DEC/VMS
- ☐ HP/MPE-V/XL
- ☐ IBM VS/VSE
- ☐ Wang VS-OS
- ☐ Other (Specify) __________

7a. Does your department have or plan to purchase a CD-ROM Drive?

- ☐ Yes, have now
- ☐ No
- ☐ Yes, plan to purchase

(Continued on Back)

Datapro Information Services Group
Delran NJ 08075 USA

8. TYPE OF NETWORK INSTALLED
Check all that apply.

- ☐ Local area network
- ☐ Wide area network
- ☐ Metropolitan area network

9. TYPE OF COMPUTER
Check all that apply.

- ☐ Mainframe
- ☐ Micro
- ☐ All of the above
- ☐ Midrange
- ☐ Workstation

10. SOFTWARE APPLICATIONS
Check all that apply.

- ☐ Accounting
- ☐ CAD/CAM/CAE
- ☐ CIM
- ☐ CASE
- ☐ Communications
- ☐ Database Management
- ☐ Decision Support
- ☐ Desktop Publishing
- ☐ Executive Information Systems
- ☐ Graphics
- ☐ Imaging
- ☐ Project Management
- ☐ Sales and Marketing
- ☐ Security
- ☐ Spreadsheet & Financial
- ☐ Planning
- ☐ Word Processing
- ☐ Other (Please specify) ____________________

11. Annual DP/MIS BUDGET
Check only one.

- ☐ Less than $50,000
- ☐ $50,000 to $99,999
- ☐ $100,000 to $249,999
- ☐ $250,000 to $999,999
- ☐ $1,000,000 to $4,999,999
- ☐ $5,000,000 or more

Fax this completed form to Marketing Planning (609-764-2813) or send this self-mailer (fold and tape with address side out).

(Fold here first)

NO POSTAGE NECESSARY IF MAILED IN THE UNITED STATES

BUSINESS REPLY MAIL
FIRST CLASS MAIL PERMIT NO. 178 DELRAN, N.J. 08075

POSTAGE WILL BE PAID BY ADDRESSEE

DATAPRO
Information Services
Group
Attention: Marketing Department
600 Delran Parkway
P.O. Box 7001
Delran, New Jersey 08075-9904

(Fold here last)

(Tape here)

Change of Address Form

Please mail or fax this form at least six weeks in advance of your address change. Make sure to include your account number so that we can process your address change quickly and accurately. (Your account number is printed on the top line of your mailing label.)

Account Number ______________________

Effective Date ______________________

New Address:

Company ______________________

Name ______________________

Address ______________________

____

City/State/Zip ______________________

New Telephone Number ______________________

For which Datapro information services do you want an address change?

☐ For all of my organization's Datapro information services

☐ For just selected Datapro information services. Specify names of the services for which you want an address change:

____

____

____

☐ I'm not sure. Please have my Customer Service Rep call me.

In case we have any questions, please complete the following:

Your Name ______________________

Your Telephone Number ______________________

Fax to Datapro Customer Service at 609-764-8953, or send this self-mailer (fold and tape with address side out).

(Fold here first)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 178 DELRAN, N.J. 08075

POSTAGE WILL BE PAID BY ADDRESSEE

DATAPRO

Information Services
Group
Attention: Marketing Department
600 Delran Parkway
P.O. Box 7001
Delran, New Jersey 08075-9904

(Fold here last)

(Staple here)

Customer Feedback—Managing LANs

Dear Customer:

I want to hear from you! Your comments and suggestions will help Datapro tailor our products to better serve your information needs. Please take a moment to tell us your ideas and return this postage paid form.

Sincerely,

Gerald J. Arcuri
Product Manager

1.

Please rate this information service on each of the following aspects:

	Excellent				Poor
Coverage	5	4	3	2	1
Ease of use	5	4	3	2	1
Format	5	4	3	2	1
Timeliness of information	5	4	3	2	1

2.

Do you have a suggestion for improving this information service?

____

____

____

3.

What topics should be added to this information service?

____

____

____

4.

How would you rate Datapro on each of the following?

	Excellent				Poor
Professionalism of sales representative	5	4	3	2	1
Ease of doing business with Datapro	5	4	3	2	1
Accuracy of billing	5	4	3	2	1
Knowledge of Help Desk consultants	5	4	3	2	1
On-time receipt of monthly update package	5	4	3	2	1

5.

How may we serve you better?

____

____

____

6.

Can we contact you for additional information?

☐ Yes ☐ No

Name ______________________________

Title ______________________________

Company Name ______________________________

Address ______________________________

City/State/Zip ______________________________

Telephone ______________________________

Fax ______________________________

(Fold here first)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 178 DELRAN, N.J. 08075

POSTAGE WILL BE PAID BY ADDRESSEE

DATAPRO

Information Services
Group
Attention: Marketing Department
600 Delran Parkway
P.O. Box 7001
Delran, New Jersey 08075-9904

(Fold here last)

Other Datapro Information Services

Datapro: The single best source for comprehensive information

For more than 20 years, leaders in business—corporate planners, sales and marketing managers, MIS professionals, and competitive analysts alike—have learned to rely on Datapro as their single best source for comprehensive information on the thousands of products and vendors that make up the computer, communications, and office technology industries.

Datapro's expert analysts continually expand and update Datapro's vast database of product and technology information to meet the specific needs of businesses. The result is a series of solutions-oriented information services covering information technology products, markets, technologies, and management issues and techniques.

Comprehensive coverage—vital to decision makers, managers, consultants, and researchers

Datapro services provide information technology users, managers, buyers, vendors, industry consultants, and sales and marketing professionals with the information and analysis they need to make better business decisions. These reports offer an authoritative and timely resource for those who need to:

- Gain a quick overview of concepts, technologies, and standards
- Evaluate or select systems, software, and solutions
- Implement and manage computer and communications systems
- Gather competitive information

Accurate and timely information on computers, communications, and office technology

Datapro offers focused information services for virtually every area of the computer, communications, and office technology industries. These services provide an accurate, timely and comprehensive source of information on products and technologies, as well as analyses of vendor strategies and markets.

In addition to extensive commentary and analysis, each service presents important information in a variety of formats—charts, graphs, and comparison tables—for quick and easy access, plus detailed reports for in-depth reviews.

Datapro information services are available on an annual basis and are updated regularly throughout the year.

Timely industry-specific reports and directories

If you're interested in a particular industry computing environment, Datapro offers services that cover banking, retailing, manufacturing, MRP (manufacturing resource planning), CAD/CAM, and factory automation.

And when you need to profile or compare information on products and vendors, Datapro's directories are current, comprehensive, and cross referenced for ease of use.

Management reports with practical value

In today's complex networking and computing environments, management issues are crucial. Datapro's "real world" management reports can provide practical guidance in planning, designing, and managing your communications, computer and software environments. Topics include management of:

- Data networks
- Voice communications
- Information Systems
- Microcomputers and LANs
- Office technology
- Applications software
- Network Management

Datapro International

Headquartered in Maidenhead, England, Datapro International publishes 11 international information services, covering communications, computers, and office systems technology. Available worldwide, the services provide extensive data on foreign-made products and vendors, as well as international standards and regulatory issues.

International distribution of Datapro information services is coordinated from three central offices, in Maidenhead, Toronto, and Singapore. In addition, Datapro International operates four sales offices in Paris, Frankfurt, Hong Kong, and Sydney.

Datapro Print Publications

COMMUNICATIONS

All the latest facts and analysis on today's data and voice communications hardware, software, and services.

Managing Data Networks

Two volumes, updated monthly.

The manager's guide to planning, designing, implementing, and maintaining a data network. Industry experts address how to plan and design organization-wide networks and internetworks, cut costs, improve performance, manage major network implementations, integrate voice and data, and prepare for ISDN.

Issues & Trends
Future Technologies
Planning
Network Project Management
Network Design
Selection & Acquisition
Installation & Maintenance
Operations Management

Architectures & Standards
Network Components (overviews)

Data Networking
Four volumes, updated monthly.

Architectures, data communications equipment, standards, networking services, internetworking and interoperability issues make up the focus of this comprehensive publication. Includes product specifications, product and vendor analyses, pricing, and market and technology overviews. Keeps users and vendors up-to-date on all areas of interest.

Technology Concepts
Vendor architectures
Standards
Multiplexers
Modems
CSUs/DSUs
Packet switching equipment
Protocol converters
Communications controllers
ISDN equipment
Leased line services
Circuit switched services
Packet switched services
LAN products
Internetworking products
Network management systems & software
Network security
Test equipment
Wiring & cabling

Datapro Reports on Data Communications International
Three volumes, updated monthly.

Covers the latest on hardware, software, and services in the international data communications marketplace. Provides reports on equipment from U.S., European, and other international manufacturers, including product specifications, analyses, pricing, and technology.

Concepts
Standards
Networks & Architectures
Communications Switches
Software
Protocol Conversion Systems
Transmission Facilities
Value Added Networks
Modems
Multiplexers
Network Management Systems
Test, Monitor, & Control Equipment

Managing Voice Networks
Two volumes, updated monthly.

A comprehensive manager's guide and training tool that provides valuable solutions and advice on creating and maintaining a cost-effective telecom network. Covers important voice issues and trends, and telecommunications concepts. Also offers advice and strategies for planning and implementing voice networking systems.

Issues & Trends
Voice Networking Concepts
Future Technologies
Management Reports
Voice Network Components
Architectures & Standards

Voice Networking Systems
Two volumes, updated monthly.

The only comprehensive source of objective information on voice products and technologies. Addresses markets and technologies, the players, and products. Includes product reviews, pricing, buying guidelines, market analyses, and user ratings.

Technology Concepts
Key/Hybrid Systems
PBX Systems
Central Office Switches
Centrex
ISDN Telephones
ACDs
Predictive Dialers
Telephone Management Systems
Voice Processing Systems
Cellular Equipment

Communications Networking Services
One volume, updated monthly.

A concise review of the networking services provided by the interexchange carriers, Regional Bell Holding Companies, and independent service providers. It addresses market trends, user ratings, and service evaluations and pricing.

Technology Concepts
Global Services
Interexchange Carriers
Switched Services
Leased Line Services
Network Access Services
Cellular Services
Packet Network Services
VSAT Services

Datapro Reports on International Telecommunications
Two volumes, updated monthly.

A unique view of the global communications industry, with details on worldwide regulatory environments, and technology overviews. Features profiles of major world vendors, international standards, and overviews by geographical region.

Regional Overviews
Standards
Vendor Profiles
Transmission Technologies
Management Topics
Directories

Datapro Reports on International Communications Equipment
Three volumes, updated monthly.

Includes technical specifications for internationally vended equipment associated with data and voice communications. In addition, communications standards, protocols, and related technology issues are presented to give a complete picture of this important market.

Technology Concepts
Standards and Protocols
Public Switches
PABXs
Key & Hybrid Systems
Specialized Switches
Communications Processors
Protocol Converters and PADs
Modems
Multiplexers
Terminals
Vendor Directory

Datapro Reports on International Communications Software
One volume, updated monthly.

Covers internationally available communications software for mainframes, minicomputers, and microcomputers. Provides in-depth profiles of the major packages available, plus overviews of data communications technology and standards, reports on network architectures, and directories of communications software available on the international market.

Concepts
Technologies
Profiles of Communications Software Packages
Communications Software Reports

Datapro Reports on Communications Alternatives
Two volumes, updated monthly.

Covers all the leading-edge transmission technologies that can be used as alternatives to commercial networks, or as competitive tools. Shows how to turn technical offerings into economic and performance advantages. Presents what all the major suppliers have to offer.

New Product Announcements
Concepts & Issues
Network Services
Network Systems
Fiber Optic Communications
Satellite Communications
Microwave Communications
Video Communications
Other Alternatives

Datapro Reports on Communications Software
One volume, updated monthly.

Directs communications managers towards software-based solutions to connectivity and network management. Includes technology overviews, reports on major network architectures, and profiles of more than 1,200 commercially available communications software packages for mainframes, midrange systems, and microcomputers. In-depth product reports highlight major software packages from leading vendors.

Technologies & Concepts
Network Architectures
Communications Software Profiles
Communications Software Reports

Datapro Reports on PC & LAN Communications
Three volumes, updated monthly.

Comparative evaluations of LAN hardware and software from the Datapro/NSTL test lab. In-depth analyses and specifications of LANs, internetworking products, PC-to-host communications products, and asynchronous communications packages provide comprehensive coverage of the volatile PC communications market.

Local Area Networks
PC-to-Host Communications Products
Asynchronous Communications Software
Electronic Mail
Microcomputer Systems
Modems/Support Equipment
LAN Hardware Evaluations
LAN Operating System Evaluations
LAN Internetworking Evaluations
LAN Software Evaluations
LAN Management Products
Technology Reports

Datapro Network Management Information Service
Two volumes, updated monthly.

A comprehensive information service that focuses on the issues, strategies, and solutions regarding one of today's biggest challenges to communications and MIS managers: network management systems.

Managing Networks
Fault Management
Configuration Management
Performance Management
Security Management
Network Planning & Design
Integrated Network Management
SNA Networks

Local Area Networks
X.25 Packet Switching Networks
Network Management Vendor Strategies
Multivendor Integrated Network Management Products
LAN & Internetwork Management Products
WAN Management Products
SNA Management Products
Telecommunications Management Products
Carrier Network Management Services
Consultants' Directory
Network Management Suppliers' Directory

Datapro Reports on International Networks and Services
Two volumes, updated monthly.

Helps managers, consultants, and vendors select, design, implement and manage private networks and use network services effectively. Provides reports on products, services and management issues concerning all aspects of networking.

Concepts
Network Options
Standards
Architectures
Applications
Local Area Networking
Wide Area Networking
Internetworking
Network Management
Network Security
Network Services
Value Added Network Services

MICROCOMPUTERS
The most comprehensive library of information on microcomputer hardware, software, and systems management.

Managing LANs
Two volumes, updated monthly.

A comprehensive information service designed for the professional who must design, implement, and manage a local area network computing environment. It is issues- and management-oriented and provides information on topics that managers must know in order to effectively plan and manage a LAN computing environment.

Technology Issues & Trends
Business Issues & Trends
Planning
Project Management
Design & Development
Selection & Acquisition
Installation & Maintenance
Operations Management
Security
LAN Technology Overviews
Computer Systems Technology Overviews
Software Overviews
Peripherals Overviews
LAN Architectures
LAN Standards

Datapro Reports on Microcomputers
Three volumes, updated monthly.

A valuable planning and selection guide to all types of microcomputer hardware and software products. Includes in-depth reports on leading microcomputer systems and business software programs. Hands-on testing provides reliable performance and ease of use reviews.

Concepts and Planning
User Ratings
Systems
Monitors
Printers
Expansion Cards
Mass Storage
Optical Storage
Communications
Scanners
Database Management Systems
Word Processing
Spreadsheets
Graphics
Integrated Software
Desktop Publishing Software
System Software
Environments/Utilities
Directory of Vendors

Datapro Reports on Microcomputers International
Three volumes, updated monthly.

Provides detailed analyses of the most popular microcomputer systems, software, and peripherals in the international marketplace. Includes surveys of U.S. and European microcomputer users, plus an international directory of vendors.

Concepts & Planning
User Ratings
Systems
Monitors
Printers
Scanners
Expansion Cards
Mass Storage
Communications Hardware & Software
Applications Software
System Software

Datapro Directory of Microcomputer Software
Three volumes, updated monthly.

Detailed profiles of over 10,000 applications and systems software packages are presented in a format that makes them easy to locate. An ideal source for comparing packages, vendors, compatible hardware systems, and prices.

Accounting
Banking & Finance
Data Communications
Data/Database Management
Education
Engineering & Scientific
Government
Graphics
Management Sciences
Manufacturing
Mathematics and Statistics
Medical and Health Care
Office Automation
Payroll and Personnel
Programming Aids
Sales & Distribution
System Programs
Utility Programs

Datapro Directory of Microcomputer Hardware
Two volumes, updated monthly.

Compares features and functions of more than 500 micros, 1,300 LAN products, and thousands of microcomputer peripherals. Also includes brief company profiles, technical specs, pricing, and options.

PCs (including portables)
Displays
Add-In Cards
Laser Printers
Impact & Ink Jet Printers
Mass Storage
Optical Storage
LAN Operating Systems and Internetworking Products
Scanners
Company Profiles

INFORMATION SYSTEMS
Covers the full range of mainframes, minicomputers, software, security, and management issues.

Computer Systems Series
Four volumes (Overviews, Systems, Peripherals, Software), each volume is also available separately, updated monthly.

The source MIS directors, CIOs, and DP managers turn to before making important purchasing and strategic decisions. Includes objective reporting and comprehensive analysis of midrange to supercomputers, peripherals, systems software, communications, and graphics devices for the enterprise.

Computer System Overviews
Computer System Reports
Computer User Ratings
Memory & Storage
Optical Storage
Printers & Plotters
Disk & Tape Drives
Scanners
Displays
Other Peripherals & Systems
Operating Systems
System Management Tools
Database Management Systems/Aids
Performance Monitors
Issues & Trends

Managing Information Technology
Two volumes, updated monthly.

Guide to issues, trends, concepts relative to information systems and communications technologies; planning, designing, and integrating IT capabilities throughout the organization; implementing major IT projects; and managing daily MIS operations.

Issues & Trends
Planning
Data Administration
IT Project Management
Design & Development
Selection & Acquisition
Third-Party Service & Support
Installation & Maintenance
End-User Support
Security
IT Overviews
Architectures & Standards

Datapro 70 International
Three volumes, updated monthly.

Provides detailed coverage of mainframes, peripherals, communications links, and graphics systems. Includes an entire volume of information on systems software; product reports on equipment from all the major worldwide computer vendors; and computer user ratings from U.S. and Europe.

Computer System Overviews
Computer User Ratings
Computer System Reports
Distributed Processing & Intelligent Terminals
Graphics & CAD/CAM
Memory & Storage
Printers & COM
Application Development Tools
Data Base Management Systems/Aids
Communications Software
Issues & Trends

Datapro International Desktop Software & Solutions
Two volumes, updated monthly.

A comprehensive information source that focuses on products in the European marketplace. The new service features desktop applications and the management of desktop computing in the corporate environment.

Concepts & Issues
Technologies to Watch
Vendor Strategies
Surveys and Testing
Systems and Server Software
Desktop Applications
Directories

Datapro Reports on Minicomputers International
Three volumes, updated monthly.

Includes the latest information on departmental processing, office systems, supermicros, and fault-tolerant systems. Provides user ratings from Europe and Japan and equipment reports from the major international manufacturers.

User Ratings
Supermicrocomputer Systems
Computers
Software
Peripherals
Communications
Vendors

Workstations & Servers
Two volumes, updated monthly

Provides the informed decisions and analyses you need to keep pace with today's increased activity in the workstation market. Market, technology and management information together with product analyses and performance measures are provided to help MIS and communications managers make informed buying decisions.

Market Issues
Technology Concepts
Standards
Management Guidelines
Peripherals
Performance
Workstation Reports
Test Results

Datapro International Workstations
One volume, updated monthly.

Concentrates on powerful desktop computer platforms and the strategic development of personal computer hardware. Features an annual strategic review of the European workstation market, based on a qualitative market research program.

Concepts & Issues
Technologies to Watch
Vendor Strategies
Surveys and Testing
Market Analysis
Systems
Directories

Datapro Reports on UNIX Systems & Software
Two volumes, updated monthly.

Understand the concepts and products that drive the UNIX operating system with this service. Issues such as open systems, ISO/OSI protocols, RISC processing, and GUIs that promise to shape the future of the market are presented. New products and strategies are analyzed as UNIX continues to evolve and new applications proliferate.

Concepts & Issues
Applications Software
Operating Systems
Communications Software
Systems/Architectures
DBMS/4GL Software
Compilers
Standards
VAR Profiles
UNIX Applications Software profiles
UNIX is a registered trademark of AT&T.

Datapro Reports on International UNIX Systems
One volume, updated monthly.

Provides up-to-date information on the concepts and products that are driving the adoption of the UNIX operating environment worldwide.

Concepts
Standards
Software: Operating Systems, Communications, Data Management, Development Tools, Applications.
Hardware: Microprocessors, Microcomputer Systems, Mid-Range Systems, Large Systems.
Management Issues

Datapro Reports on Information Security
Three volumes, updated monthly.

Solve the toughest information security problems using this practical service. Created to help users take effective steps to control viruses as well as secure microcomputers, networks, mainframes, and physical sites. Get the up-to-date facts about security technology and government regulations.

Microcomputer Security
Network Security
Disaster Avoidance
Host Security Software
Planning
Risk Analysis
EDP Auditing
Physical Security
Standards, Policies, & Regulations
Specialized Systems & Applications
Suppliers & Consultants
Concepts & Issues

Datapro Reports on Software International
One volume, updated monthly.

Details over 150 systems software packages available domestically and overseas. Provides objective information on the major software packages for mainframes, minis, and supermicros. Includes reports on the availability of international software products and a directory of international suppliers.

Software Concepts & Trends
Application Development Tools
Communications Software
Data Base Management Systems/Aids
Performance Monitors & Security Systems
Systems Enhancements
Utilities
Data Management
Operating Systems

Application Development Software
One volume, updated monthly.

Covers the leading applications development products for midrange and mainframe computers. Product profiles serve as valuable aids in matching state-of-the-art features to your needs. In-depth analyses provide overviews of current and future market trends.

Concepts
Computer Aided Software Engineering (CASE)
Decision Support Software
Project Management & Executive Information Systems (EIS)
User Interface Management
Expert & Knowledge-based Systems
Fourth Generation Languages
Query & Report Writers
Geographical Information Systems

Datapro Management of Applications Software
Two volumes, updated monthly.

Guides DP and application development managers through every phase of applications software acquisition and development. More than just a planning tool, this service offers innovative ideas and helpful advice to help you control development and maintenance costs, improve productivity, and extend the life of your software, making it more efficient and cost effective.

Software Development Concepts and Techniques
Planning & Cost Justification
Software Design
Guide to Application Development Products
Selection & Acquisition
Software Production
Reliability
Installation & Testing
Maintenance
Performance Measurement
Emerging Trends
Directories of Outside Resources

Datapro Directory of Software
Three volumes, updated monthly.

Covers all varieties of commercially available midrange and mainframe software. Profiles more than 11,000 packages on the basis of features, price, compatibility, date of introduction, installed base, and other characteristics.

Accounting
Banking & Finance
DP Center Management
Data Communications
Data Management
Education
Engineering & Scientific
Insurance & Real Estate
Language Processors
Management Sciences
Manufacturing
Medical & Health Care
Office Automation
Payroll & Personnel
Programming Aids
Sales & Distribution
System Programs
Utilities

OFFICE TECHNOLOGIES
Objective, comprehensive advice and guidelines for making your office or electronic publishing operation more efficient and productive.

Datapro Management of Office Automation
Two volumes, updated monthly.

Offers the latest thinking on planning, designing and managing office technologies such as fax, copiers, PCs, and electronic mail. Includes overviews of various office technologies and systems, practical guidelines for decision-making, planning models, and proven advice from experts in the field.

The Current Office
Evolving Office of the Future
Office Systems Development
Equipment Evaluation & Selection
Financial/Legal Management
Personnel Management
Facilities Management
Operations Management
Systems Management
Automated Office Software
Consultants & Suppliers
Glossary

Datapro Reports on Office Automation
Three volumes, updated monthly.

Focuses on computer-based systems and equipment used in the total document management environment of an organization. Provides a wealth of detail about the many options for handling end-user computing technology in the office environment.

Office Automation Issues
User Surveys
Integrated Systems
Local Area Networks
Electronic Mail & Messaging Systems
Office Automation Software
Microcomputers
Printers
Micrographics
Computer Assisted Retrieval
Scanners
Electronic Typewriters
Facsimile
PBX Systems
Copiers
Vendor Directory

Datapro Reports on Office Automation International
Three volumes, updated monthly.

An essential information source for the OA professional. Includes reports on international manufacturers of office systems, local area networks, facsimile systems, and office software.

Office Automation Issues
Market and Technology Overviews
Integrated Systems
Networked Systems
Electronic Mail & Messaging Systems
Workstations
Electronic Typewriter Systems
Printers
Office Automation Software
Facsimile Systems
Image Processing & Micrographics

Datapro Reports on Electronic Publishing Systems
Two volumes, updated monthly.

An in-depth sourcebook for anyone wishing to use the technology of desktop publishing. Includes comparisons of equipment and costs, case studies, and results of software testing. Provides up-to-date information on rapid advances in this dynamic field.

Publishing Fundamentals
Concepts & Issues
Strategies
Case Studies
Dedicated Systems
Hardware Systems
Monitors
Publishing Software
Graphics Software
Typesetters
Printers
Scanners
Electronic Delivery Media
Related Systems & Standards
Vendor Directory

Datapro Office Products Evaluation Service
Two volumes, updated monthly.

The industry insider's guide to copiers, electronic typewriters, and facsimile machines. Provides comprehensive specifications and pricing, but goes beyond to examine functions and performance of products. Includes hands-on test summaries, user ratings, and training guidelines.

Training Perspectives
Copier Comparison Guides
Copier Product Reports—Low-, Mid-, High-Volume
Commercial Copier Pricing
Electronic Typewriters
Facsimile
Laser Printers
Related Technology
User Ratings
NOMDA Bulletin

Datapro Reports on Document Imaging Systems
Two volumes, updated monthly.

Provides the only complete information source on document imaging systems technology, manufacturers, and products. Includes complete and comprehensive specifications and information on image processing hardware and software and a newsletter on late-breaking announcements in the industry. Also includes valuable management information that presents the latest thinking on how to get the maximum productivity and efficiency from your document imaging system.

Management Issues
User Perspectives
Document Imaging Systems
Document Imaging Software
Input Technologies and Products
Output and Retrieval Technologies and Products
Storage Technologies and Products
Related Technologies
AIIM Bulletin

INDUSTRY AUTOMATION

Datapro Reports on Banking Automation
One volume, updated bimonthly.

Combines product analysis and user ratings with authoritative and up-to-date information needed by banks, savings and loans, and thrift institutions. For bank DP managers, branch automation managers, and operations officers who need to stay current on the latest topics and technology in this dynamic business segment.

Technology Trends
Teller & Platform Equipment
Automated Tellers
Image Processing Equipment
Software & DP Services
Item Processing
Lockbox & Backroom Equipment

Datapro Reports on Retail Automation
One volume, updated bimonthly.

Complete information on automated equipment and software for general merchandising, supermarket, and major hospitality organizations. Product comparison tables and reports point out the advantages and restrictions of hardware and software products.

Technology Trends
Controller-Based Retail System
Electronic Cash Registers
Credit & Payment Systems
Specialized Retail Systems
SKU/UPC Marking & Reading Equipment
Retail Software & Services
Vendors

Datapro Manufacturing Automation Series
Management and Planning, updated bimonthly.
Manufacturing Information Systems, updated bimonthly.
CAD/CAM/CAE Systems, updated bimonthly.
Factory Automation Systems, updated bimonthly.
News and Perspectives, monthly newsletter.

Provides product analyses, surveys, market trends, and users' experiences relative to the entire spectrum of manufacturing automation practices. Includes strategic planning guidelines, evaluations of micro-based software, and product reviews with specifications to help you automate your manufacturing environment. The newsletter complements the other volumes by providing industry insights, case studies, interviews, and timely news analysis.

Technical Workstations
Multiuser CAD Systems
Microcomputer-based CAD Systems
Communications
Manufacturing Information Systems
Microcomputer-based MRP Products
Issues and Technologies
Industrial Control
Planning Guidelines
Automated Materials Handling
Data Collection
Graphic Peripherals
Robotics
Specialized Technologies
Operations Management

Datapro Reports on Marketing Information Systems
One volume, updated quarterly. Eight newsletters per year.

Presents important technological developments and industry-specific case studies critical to successful database marketing operations. For marketing, sales, and MIS managers who need a comprehensive, objective sourcebook on the latest management techniques, proven strategies, and available products and services.

Market/Customer Database Systems
Sales Analysis/Productivity Systems
Marketing Research Systems
Telemarketing Systems
Order Fulfillment Systems
System Designs/Solutions
Special Applications Systems
Suppliers & Services
Issues & Opportunities

Datapro Reports on Service & Support
One volume, updated monthly.

Contains a comprehensive directory of service vendors, side-by-side comparisons of services and pricing, in-depth competitive evaluations of the major service providers, and user perspectives: satisfaction ratings and buying plans.

Industry Surveys & Market Overviews
User Ratings & Perspectives
PC LAN Support
Software Maintenance & Support
Education & Training
Negotiating Service Contracts
Systems Integration
Hardware Maintenance
Network Services Support
Systems Integration
  Directories
  Vendors
Service Providers by Region
Services & Support Announcements
Outsourcing

Datapro Electronic Products

Datapro Software Finder
Updated quarterly.

Datapro's authoritative software directories on CD-ROM, Software Finder helps you save time identifying and selecting software products, and make better-informed purchase decisions. Profiles over 19,000 business and professional software packages for micro, midrange, and mainframe systems. With nine different search criteria, including key words, you can quickly identify products that meet very specific requirements. Profiles give you all the basic facts at the touch of a key: price system compatibility, memory requirements, network support, vendor contact information, a brief description, and more.

Datapro Competitive Edge in Communications
CD-ROM-based service for companies in the communications industry---to help sales and marketing managers track

and stay out in front of competitors. Provides in-depth profiles of key communications vendors, detailed product analyses, including strengths and limitations, product comparison columns listing technical characteristics, and pricing data on more than 1,300 communications products and industry announcements. Includes separate CD-ROM products on voice, data communications, and LANs.

Computer-Based Training
Designed to provide a cost-effective alternative to more formalized learning. Datapro's Computer-Based Training (CBT) courses are developed under the expert guidance of experienced industry leaders so you gain insight into real-world problems.

Datapro Faxline
This facsimile transmission service provides subscribers with the latest-breaking news on copiers and fax machines. New product announcements are faxed within 48 hours of the announcement, and trade show reports are sent within one week of the show.

CASEBASE
A diskette-based menu-oriented software evaluation tool that covers all CASE products, from analysis & design products, to technical CASE products, to reengineering. It offers side-by-side comparisons of more than 250 CASE products from more than 150 vendors, a detailed description of CASE products, a database of new product announcements, and enhancements.

Datapro Custom Information Services

Database Products
Most of Datapro's print products are also available in ASCII format. Especially valuable to vendors who distribute information via wide-area networks.
- Competitive Products Database
- Designed specifically for sales and marketing managers, product planners and product managers, this electronic service offers detailed reports, updated weekly, on more than 300 vendors and 750 products. Profiles of company financial positions, sales and marketing strategies, sales channels, terms and support policies are provided.
- NewsBASE
- An ideal tool for searching, manipulating, and analyzing events in the computer and communications industries. Updated daily.

Handbooks

(Competitive and Vertical Market)
Easy to carry and use, these customized handbooks give salespeople a portable tool for accessing the latest information on competitors' products, technology and industry trends, user ratings, and more.

On-Call Consulting
When you need quick, knowledgeable answers to product and technology questions, this phone-in service offers quick response—usually within minutes.

Surveys
Includes both Customer Satisfaction and User Ratings. The former alerts companies to the strengths and limitations of their products as seen by users. User Rating Surveys provide in-depth information across a range of hardware and software categories.

Northern Business Information
As the world's leading supplier of telecommunications research, Northern Business Information offers a vast array of market and competitive assessment reports and services that help your company manage and reduce the risks associated with this ever-changing market.

National Software Testing Laboratories (NSTL)
As the largest independent testing and evaluation facility in the microcomputer industry, NSTL can consult with your company in helping design, test, and enhance your products. ■

Information Request

For detailed information on any of Datapro's information services, just check your area of interest. Return this postage-paid flyer today.

Name ______

Title ______

Company Name ______

Address ______

City/State/Zip ______

Telephone ______

Fax ______

Communications

- ☐ Managing Data Networks
- ☐ Data Networking
- ☐ Reports on Data Communications Intl.
- ☐ Management of Telecommunications
- ☐ Reports on Telecommunications
- ☐ Reports on Intl. Telecommunications
- ☐ Reports on Intl. Communications Equipment
- ☐ Reports on Intl. Communications Software
- ☐ Reports on Communications Alternatives
- ☐ Reports on Communications Software
- ☐ Reports on PC & LAN Communications
- ☐ Reports on Network Management Systems
- ☐ Network Management
- ☐ Reports on Intl. Networks & Services

Information Systems

- ☐ Computer Systems Series
  - Overviews
  - Reports
  - Peripherals
  - Software
- ☐ Datapro 70 Intl.
- ☐ Managing Information Technologies
- ☐ Management of Applications Software
- ☐ Workstations and Services
- ☐ Reports on Minicomputers Intl.
- ☐ Reports on UNIX® Systems & Software
- ☐ Reports on Intl. UNIX Systems
- ☐ Reports on Information Security
- ☐ Reports on Software Intl.
- ☐ Application Development Software
- ☐ Directory of Software
- ☐ Reports on Service & Support

Office Technologies

- ☐ Management of Office Automation
- ☐ Reports on Office Automation
- ☐ Reports on Office Automation Intl.
- ☐ Reports on Electronic Publishing Systems
- ☐ Office Products Evaluation Service
- ☐ Reports on Document Imaging Systems

Industry Automation

- ☐ Reports on Banking Automation
- ☐ Reports on Retail Automation
- ☐ Manufacturing Automation Series
- ☐ Reports on Marketing Information Systems

Microcomputers

- ☐ Managing LANs
- ☐ Reports on Microcomputers
- ☐ Reports on Microcomputers Intl.
- ☐ Directory of Microcomputer Software
- ☐ Directory of Microcomputer Hardware

Electronic Products

- ☐ Software Finder (CD-ROM)
- ☐ Competitive Edge (CD-ROM)
- ☐ Computer-Based Training (CBT) (Diskette)
- ☐ FAX Track
- ☐ Datapro Faxline
- ☐ CASEBASE

Need help choosing the right information service?
Call the Datapro Help Desk at 1-800-328-2776, ext. 2470.
Fax to Marketing Department at 609-764-2812,
or send this self-mailer.

DATAPRO
Information Services Group

(Fold here first)

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 178 DELRAN, N.J. 08075

POSTAGE WILL BE PAID BY ADDRESSEE

DATAPRO

Information Services
Group
Attention: Customer Service
600 Delran Parkway
P.O. Box 7001
Delran, New Jersey 08075-9904

(Fold here last)

(Tape here)

The Evolution of LAN Cabling

In this report:

Datapro Summary

The incredible proliferation of LANs within the last decade has created many cabling options for these networks. Continuing technical refinements in the data communications field have improved the information carrying capacity of existing cable technology. The use of fiber optic technology has now become cost-effective and increasingly commonplace. Both users and manufacturers have fostered, as well as benefited from, the development and implementation of formal cabling standards. When armed with an awareness of existing standards and evolving trends, an appropriate LAN cabling strategy can be determined for any given application.

Cabling at the Inception of LANs

Many buildings now carry sharp reminders of the ways in which communications cabling was accomplished in years past. The cabling that still exists, often unused, within many buildings reflects how an unplanned, unmanaged entity can spin out of control. From the initial architectural design through the completion and occupancy of most buildings, telecommunications wiring was rarely considered, except for the minimal space allowed for the local telephone company. Unfortunately, this space was often located near electrical systems and was therefore susceptible to electrical interference.

Telephone companies widely used 25-pair cable for each telephone; this was nearly one-half an inch thick in diameter and very labor intensive to install and terminate. The telephone companies were sometimes fortunate enough to install their cable before interior construction was complete. This allowed their installers easy access to conduits, underfloor ductwork, and spaces between walls and above ceilings. Installers of data communications cabling were seldom as fortunate. While the selection of telephone cable (which was suitable for a multitude of telephone systems) was left to the local telephone company, building tenants often did not select their computer cabling until after they had occupied the building. This was, in part, due to the fact that the cable type and specifications hinged on the selection of the computer itself.

Proper cable installation at this point was difficult, expensive, and rare. Data communications cabling often contended with telephone cabling for any remaining conduit or duct space. Cabling for data terminals draped down from ceilings, along baseboards, and even across floors. Compounding this problem was the fact that the use of computer terminals was on the rise. Each occupant of the building was usually provided with a telephone and its associated cable from the date they occupied the facility. The initial data terminal cabling, however, was only installed for the small percentage of the employees accessing the company's computer. This user population grew steadily to the point that as much as

—By Bernard A. O'Donnell
Director of Communications
Office of Information Technology
of the State of Connecticut

half of the staff (or more) required system access. Cables were run on an ad-hoc basis to each new terminal and to any existing terminals that were relocated.

Many unused, unmarked cables were abandoned, further complicating the wiring process and choking conduits. When the computer system itself reached capacity and replacement was imminent, the user frequently found that with the new system came new cabling requirements. Unless conduits and underfloor ducts were filled to capacity, old and unused computer or telephone cable was rarely removed. The labor costs of removing the cable were prohibitive, and the possibility of damaging ceilings, walls, or existing live connections was too great. The lack of order and management of the cabling system caused still more cables to be run, since it was easier to install new cable than to trace and test existing, unused cables for re-use. Companies using multiple vendors' computer systems in different departments routinely found that each system used a proprietary data communications scheme and a different cable type.

Although cabling standards and cable management systems exist today to minimize the grim aspects of the aforementioned scenario, all too many terminals and local area networks are still cabled in the same fashion. Proper planning, adherence to industry standards, and ongoing management of cabling system are imperative to avoid the expensive, inefficient "rat's nest" that can result from ad-hoc cabling.

LAN Cabling Requirements

As local area networks were developed and evolved, a wide variety of options evolved for the physical layout of the network; network topology; bandwidth; channelization; and cable types. Within these options exist advantages and disadvantages based on the combination of the physical and logical aspects of a particular local area network architecture.

Topologies

The primary wiring topologies for the physical connection of local area network devices are the bus, ring, and star. The bus topology (see Figure 1) utilizes a single cable to which all network devices attach. Network devices, or nodes, are connected by "tapping" into the communications bus at the location of each node. This facilitates the initial installation of a network, but additions to the network can sometimes be difficult, depending on the accessibility of the bus cable. The communications bus is "terminated" at each end with a device that prevents signals along the cable from canceling each other out or otherwise interfering with the normal operation of the network.

Any cable break along the communications bus, or the removal or failure of a termination, will cause the entire network to fail. Some bus-based networks operate by allowing each station on the network to broadcast an addressed message to all network devices simultaneously. Even though the integrity of the cable may be preserved between most or all of the nodes, no communication can take place between any stations unless the entire bus is undamaged and properly terminated. Network troubleshooting, maintenance, and the addition of new devices often dictates that the entire network be taken out of service until such work is completed.

With the ring topology (see Figure 2), the network cable is configured as a loop to which all network nodes attach. Each network device is actually inserted into the ring and receives, and either processes or retransmits, each message on the network. Based on the address associated with each message, the station determines if it is the intended recipient of the message. If the station is not the intended destination for the message, it repeats the message to the next device on the network. All transmissions are repeated throughout the network, station by station, as each network device reviews the address of each message.

As the number of network devices increases, this serial transmission scheme can impact the network's response time. Since the network depends on each node to retransmit each message, a cable failure at a single point within the ring can cripple the network.

The star topology (see Figure 3) utilizes a central controller to which all network devices connect. The controller performs a "traffic manager" function by establishing connections on the network as required by the individual network devices. The central controller in a star network can become congested with traffic as the number of network nodes increase. It is also the single point of failure within the LAN. Since the controller (or server) contains a number of active components combined with operating software, it is much more prone to periodic failure than the network cabling itself.

Variations and combinations of these basic network wiring topologies have also evolved. Controllers within multiple star networks can be interconnected, a "tree" topology exists which consists of interconnected bus networks, and LANs consisting of multiple rings are common. Each topology has specific strengths and weaknesses, and LAN cabling strategies have developed in such a way as to derive the best features from each topology, and combine

Figure 1.
Bus Topology

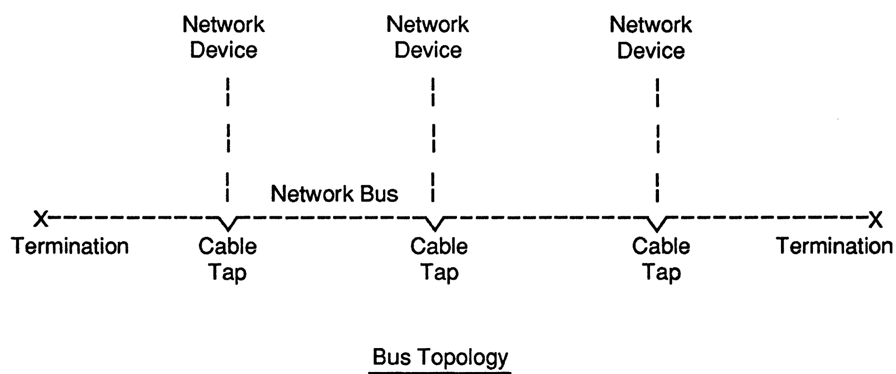
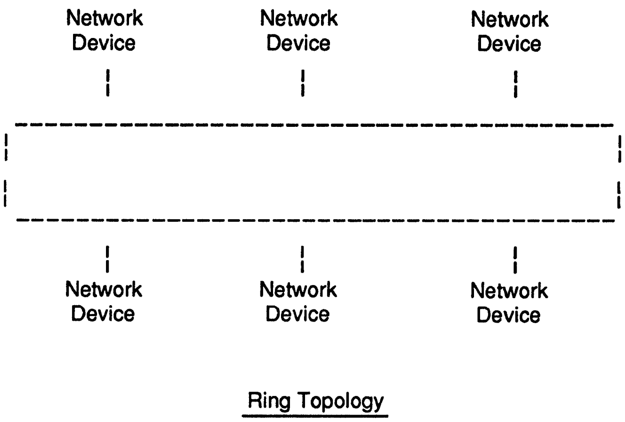


Figure 2.
Ring Topology



them into a common methodology that supports a number of different network access methods.

This methodology consists of providing a "home-run" cable connection from each network device to a common wiring hub (see Figure 4). In a small network, this network arrangement can consist of a single hub co-located with a server or computer room. In a large network, there can be a number of these hubs, each serving network devices within a specific section of a building (generally stations within an approximate radius of 250 feet from the hub). This borrows two significant advantages from the star topology: additional stations are simply added to the network by cabling to the closest available hub; and any cable-specific network problems can be easily isolated and resolved. The hubs themselves, in their most basic form, are no more than connection points for the wires that connect each network device. The cable from each device is accessible at the hub to allow interconnection to other LAN components. Although physically wired in a star configuration, the LAN components that are interconnected to the hub determine the operational, or logical, topology of the LAN.

The network device ports can be interconnected to form a bus, ring, or star network. This arrangement allows multiple LANs, even those utilizing different logical topologies, to coexist within the same cabling system. It also eliminates the need to recable in the event that the overall network topology changes. In a network with multiple hubs, the individual hubs are cabled together to form a single network.

Baseband or Broadband

The majority of local area networks installed today are known as "baseband" networks. A device attached to a baseband network uses the entire communications channel, or bandwidth, with each data transmission. Baseband networks use digital communications techniques, and each device is allowed its own turn to transmit on the network. Various baseband LANs typically operate at speeds of 1M, 4M, 10M, and 16M bps. The medium that carries these digital signals is either coaxial, fiber-optic, unshielded twisted-pair, or shielded twisted-pair cable.

A higher capacity, though less prevalent, type of network is known as a "broadband " network. Broadband LANs are similar in nature to cable-television networks that distribute many channels simultaneously over multiple frequency bands. Rather than allow a single network device to utilize the entire capacity of the network, a broadband network essentially consists of multiple logical baseband channels that can be used simultaneously for different baseband services. The device that attaches to the network cable, typically of coaxial construction, uses a modem tuned to a particular network channel in order to establish communications. The digital data signals are modulated in order to be carried over the cable as analog radio-frequency signals.

The channel frequencies of a broadband network often carry additional services along with typical LAN traffic. Video, voice, energy management, security, radio, and other data services can be carried since these networks generally offer a total bandwidth of 300 to 400MHz, with each baseband channel usually allocated 6MHz, depending on the service to be carried. Broadband networks carry signals in both directions at the same time, either by dividing the bandwidth of a single cable in half and utilizing two-way transmission on that cable, or through the use of two cables. The coaxial cable and components used for broadband networks are fundamentally the same as used in the cable-television industry. With the number of active components, such as modems and amplifiers, and the relatively expensive (and rigid) coaxial cable required to implement a broadband network, it is more expensive and difficult to install and maintain a broadband network than a similar-sized baseband network.

Broadband networks are not typically used for single-service applications. The broadband network has found its widest use in campus environments with requirements for multiple services and the need for future capacity over a widely dispersed area.

Twisted-Pair and Coax

Twisted-Pair Wire

As LANs evolved, more cabling options became available to the user. Early Ethernet networks used thick coaxial cable exclusively; today, Ethernet LANs are implemented with both thick and thin coaxial cable, fiber-optic cable, or

Figure 3.
Star Topology

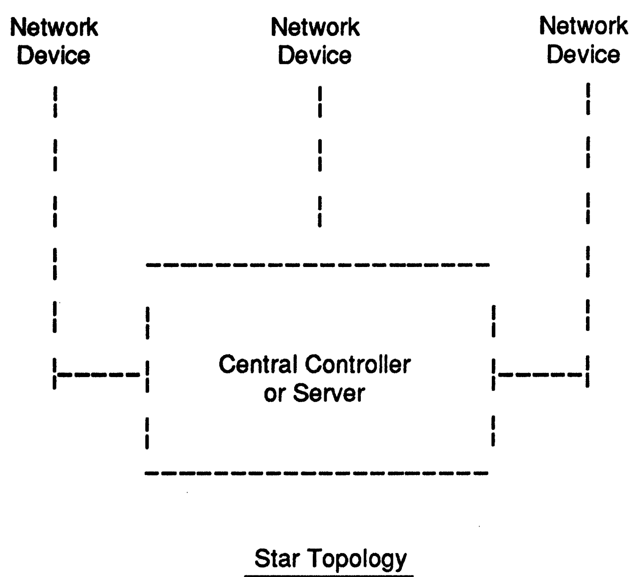
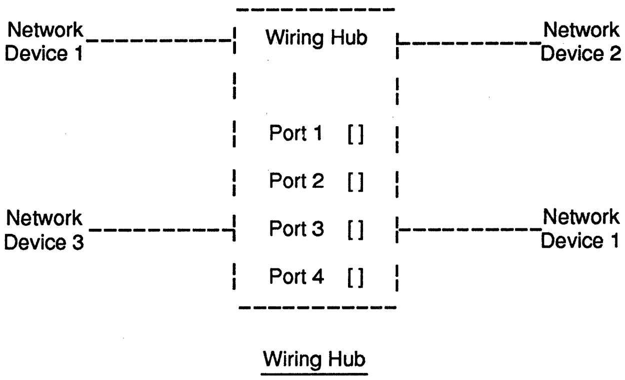


Figure 4.
Wiring Hub



twisted-pair copper cable. With the approval of the IEEE 802.3 10BASE-T specification, an upsurge in Ethernet installations using unshielded twisted-pair (UTP) is under way. Token-ring networks are now being widely installed using unshielded twisted-pair cable, whereas only shielded twisted-pair (STP) cable was considered suitable for token-ring LANs a few years ago.

These options developed as cabling options for other data communications systems improved. These improvements occurred when some vendors began to offer products that allowed data terminals to operate over standard telephone cable—unshielded twisted-pair. The timing of the release of many of these products coincided with the acquisition of inside telephone wiring by telephone company customers as a result of the deregulation of the telephone companies. Data communications managers found that they could utilize the spare capacity of this existing network within their buildings to connect their computer terminals. Since the telephone wiring in most buildings already ran to virtually every desk and work location, this offered significant cost and flexibility advantages.

The most noteworthy disadvantage of this approach was the initial lack of support that the major computer vendors provided. Users found that many vendors discouraged the use of telephone cable. These vendors maintained that UTP was frequently the source of terminal outages. Established standards for the use and testing of UTP wiring pertained only to its use within the realm of voice communications. Users were frustrated by vendors' reluctance to support this "new" type of cabling for data communications, and vendors were disenchanted with the volume of service outages attributable to improper cabling. Success and horror stories abounded within the data communications community concerning the use of UTP telephone cable as a viable means of connecting terminals. In reality, the failures were due primarily to unsuitable grades of cable or improper installation practices. As more companies entered the market with products for UTP wiring systems, the level of support improved.

Eventually, the major computer vendors noticed that user acceptance of these products was on the rise, and they began to offer and support products for unshielded twisted-pair wiring systems. As these large vendors began to implement and support UTP wiring systems for their customers, they established their own criteria for the installation and use of the wiring. Each vendor discovered the limits of each wiring type for use with their own systems. Limitations in distance, number of connection points, and other specific boundaries were established based largely on the electrical characteristics of the type of cable used.

Coaxial and Twinaxial Cable

The most popular cable types used for data communications prior to the widespread use of unshielded twisted-pair wiring were coaxial, twinaxial, and 25-conductor RS-232 cable. All are significantly more costly to purchase and install than unshielded twisted-pair telephone cable, and RS-232 cables are severely limited in the distances they can reliably carry certain signals. Terminals using RS-232 cables are easily adapted to twisted-pair wiring either as a direct substitution for the former cable type, or via the use of inexpensive modems or line drivers to boost signals, depending on the distances involved.

Coaxial and twinaxial cables are adapted to twisted-pair wiring through the use of a transformer-type device called a "balun." The balun adapts the "balanced" twisted-pair wiring to the "unbalanced" coaxial or twinaxial cable. (In this context, the term balanced signifies that the individual signal wires within a cable exhibit identical electrical characteristics in reference to ground. Wires within the balanced twisted-pair cable that are required to interface to the coaxial or twinaxial cable are not grounded. Coaxial and twinaxial cables are known as unbalanced cables because the cable shield, which acts as a signal wire, is grounded.) The balun acts as an interface to match the differing electrical characteristics of twisted-pair cable to the coaxial and twinaxial cables. Various other names for baluns, including media filter and twisted-pair adapter, are common.

Cabling Systems and Standards

Cabling Systems

Within the past decade, information systems users have come to demand that new products exhibit a standards-based approach to systems interconnectivity. In response to this demand, a number of vendors have developed and marketed complete wiring systems that include components suited to a number of different types of telecommunications applications and systems, encompassing voice, data, and image communications. IBM, AT&T, Digital Equipment Corp., AMP, Northern Telecom, and others have entered this market, which had previously been almost exclusively the territory of smaller, niche market suppliers such as Mod-Tap, Nevada Western, Ortronics, and others.

While the smaller firms acted as pioneers for these products, and often contribute to the larger firms' product line as component suppliers, each of the large information systems vendors has helped to refine and gain acceptance for a standard approach to building wiring. The IBM Cabling System was initially developed to support the IBM Token-Ring Network and offers a number of different cable categories designed to meet current and future cabling requirements. These categories of cable are designated as Type 1, Type 2, Type 3, etc. Each cable type has a different physical construction and is designed for a specific range of applications.

AT&T's SYSTIMAX PDS system consists of twisted-pair and fiber-optic cable that can be adapted to meet the requirements of virtually any major system. Digital's DEConnect system also includes support for video signals via coaxial cable, while AMP's system allows the use of interchangeable, snap-in faceplates to allow the cabling system to support rapid system reconfigurations. Northern Telecom is emphasizing its commitment to its cabling system by certifying and warranting its capability to maintain operation within predetermined specifications. Many other suppliers exist, and most of their systems provide a similar level of functionality.

Cabling Standards

Even though the use of twisted-pair and fiber optic cabling systems is becoming commonplace, equipment manufacturers still find themselves unable to anticipate the electrical and physical specifications of their customer's cabling systems, to which their products must connect. One customer's building might be wired with three-pair cable to each desk, while another will contain four-pair cable. Different electrical properties exist across the various types of twisted-pair wiring that is available. Some end users are implementing one type of fiber optic cable while others choose a different class of fiber. For the equipment vendor, this increases the complexity of introducing new products, and makes life difficult for its installation and service staff.

In response to these problems, the Electronic Industries Association (EIA) and the Telecommunications Industry Association (TIA) began development of a telecommunications wiring standard. The members of the association and other interested parties worked together to come to a consensus on a standard approach to building wiring that would serve a wide variety of telecommunications systems.

From that effort came the following standards:

- Commercial Building Telecommunications Wiring Standard (EIA/TIA-568)
- Commercial Building Standard for Telecommunications Pathways and Spaces (EIA/TIA-569)
- Residential and Light Commercial Telecommunications Wiring Standard (EIA/TIA-570)

The Commercial Building Telecommunications Wiring Standard addresses specifications for the standard application of twisted-pair, coaxial, and fiber optic cable. It includes recommended topologies, distance limitations, installation practices, and hardware specifications.

The Commercial Building Standard for Telecommunications Pathways and Spaces mandates a standard approach for implementing and utilizing the conduits, cable trays, ducts, manholes, and other paths and spaces in which cable is installed.

The Residential and Light Commercial Telecommunications Wiring Standard is similar in content to the Commercial Building Telecommunications Wiring Standard, but addresses the specifics of residential wiring.

A separate standard addressing the particulars of industrial-building telecommunications wiring may be published by the EIA/TIA in the near future. Specifications within the EIA/TIA standards support 10BASE-T and other physical options of the 802.3 LAN standards developed by the Institute of Electrical and Electronic Engineers (IEEE) as a part of their overall Project 802 standards effort regarding local area networks. The 10BASE-T specification provides for operation of a 10M bps baseband Ethernet LAN over unshielded twisted-pair wiring, and is one of the most stringent data communications applications for UTP cabling.

The EIA/TIA standards also support other IEEE standards, such as IEEE 802.5 (token-ring); the American National Standards Institute's (ANSI) Fiber Distributed Data Interface (FDDI); and the physical aspects of the Integrated Services Digital Network (ISDN) standards. Anyone planning to implement a new building wiring system should purchase and study copies of the commercial standard and the pathways standard. Both are available through the EIA at 2001 Eye Street N.W., Washington, D.C. 20006; telephone (202) 457-4900. These are considered the standards for telecommunications wiring within the industry today, and suppliers of building wiring systems and components are complying with the EIA documents at every opportunity.

Planning, Implementation, and Operational Considerations

Today, the cabling that supports local area networks and other telecommunications services within a building or campus has evolved to become an architected component of that facility. Ideally, the end user is involved during the design phase of the building construction project and can participate at each decision point that might later affect the ability of the structure to support a given cabling system. In the event that this is not the case, the architect's and building owner's adherence to a standard approach to the cabling system can help ensure that the tenant's LAN and other telecommunications services will be adequately supported. Buildings are often constructed without knowing who the tenants will be, much less what their LAN cabling requirements will be.

Implementation of cabling systems that support the EIA/TIA wiring standards is becoming commonplace. Architects, engineers, and contractors have developed an awareness of these standards-based cabling systems and often use them as a baseline requirement for new office facilities. The cabling systems themselves have become as much a part of the building systems as the plumbing or electrical systems. This is in contrast with a time when cabling was considered more a part of the computer system it supported rather than the building. Although either option can be appropriate for a given situation, long-range considerations weigh in favor of adopting a single cabling system as a part of any new or renovated facility.

A critical factor to consider when designing a cabling system for a medium- to large-sized office facility is the opportunity for change. A single-tenant building might later become a multi-tenant office, or vice-versa. Wiring systems must be designed to allow either interconnection or dissection of various segments of cable. Wiring initially installed to serve a single user could serve four separate tenants at a later date. An adequate number of equipment rooms with appropriate electrical service, security, access, environmental conditions, and floor or wall space must be considered with an eye toward the future.

LAN access should be readily available to any location within the facility, and an adequate number of prewired LAN access jacks for future use should be part of the LAN cabling plan. This provides flexibility for relocation of existing network devices as well as addititonal staff. Since future expansion is difficult to predict, it is wise to provide LAN access to virtually every possible future location. Office copiers have not typically required any wiring other than electrical service, but as they become common as network devices on LANs, wiring even those locations will become routine.

Sharing the Facility

As a user begins the physical aspects of implementing the LAN, he or she may find that other parties plan to use a conduit, wall, access sleeve, or cable that was originally anticipated to be available exclusively for the LAN. Many telephone company installers are now very restricted in the level of freedom they can exercise in choosing a point to mount and terminate incoming cables and equipment. The restrictions typically dictate that the telephone company terminate the incoming service cables and mount the equipment at a single entrance point to serve the entire facility. This entrance point is generally as close as possible to the point that the cable enters the building.

The necessary distribution of cabling within the building is considered to be the responsibility of the building owner, and the local telephone company does not become involved unless the building owner contracts with it separately for the internal cabling of the building. In addition to the issue of competition with the telephone company for wall space or electrical service, the LAN implementer may want to consider the need to provide connectivity to the telephone company demarcation point in the event that an external data circuit is required to bridge to another LAN, access a host computer, or provide dial-up access to the LAN.

In adddition to the telephone company, others contending for conduit space, power, and equipment space can be installers of telephone systems, other data communications systems, security, video, paging, electrical, or radio systems. Proper coordination among the installation efforts for these systems is not only required for the aforementioned reasons, but also to ensure that no adverse effects result from any undesireable interaction between them. Of all the systems within a building, the LAN is usually the most susceptible to any electromagnetic, radio frequency, or electrical power interference.

Comparing Twisted-Pair

Shielded twisted-pair cable may become a necessity in order to protect the network from other signals and electrical noise. The additional cost of STP cable can be insignificant compared to the magnitude of the operational problems that may ultimately exist within a harsh electromagnetic environment. Proper cable installation and grounding techniques become extremely important when shielded cable is used, in order to prevent the introduction of spurious signals onto the cable shield. Adherence to the LAN manufacturer's installation specifications is critical to avoid any electrical interaction between systems.

Although the initial goal of many twisted-pair wiring products was to allow the re-use of existing telephone cabling within buildings, this is not the most common means by which LANs using twisted-pair cabling are implemented. Different grades of twisted-pair cabling exhibit varying electrical properties that determine their suitability for transmission of an electrical signal. Some factors that are key in determining a cables' data transmission abilities for a LAN are listed below.

- Wire gauge (diameter)
- Insulating material
- Insulation thickness
- Number of wire twists per foot
- Electrical resistance
- Capacitance
- Signal attenuation at a given frequency or data rate
- Balance properties

These qualities are generally specified by manufacturers of equipment to be connected to the cable, but ideally, a cable should be suited to meet the highest level of functionality eventually required within a building. Standard telephone cabling may be suitable for certain types of low speed data transmission, but local area networks operating at multi-megabit data rates often require better grades of twisted-pair cable.

Cable Testing

Before making the decision to use, or not use, existing cable to wire a LAN, some testing and analysis should be done. Testing of the in-place cable should be done in order to determine that it meets the necessary manufacturers' specifications for the LAN being installed. In addition to electrical testing, existing cable should be tested up to the specifications of any anticipated applications. This allows the user to determine the useful life of that specific grade of cable in that environment. If the existing cabling is electrically suited to the application at hand and any new applications are far in the future, it might be unwise to install new cabling since the state of the art in either the application or the cabling systems might change in the interim.

For testing existing cable, a guideline such as Digital's *Unshielded Twisted Pair Ethernet Wiring Installation and Characterization Guide* should be used. In addition to electrical tests, existing cabling should meet other criteria in order to be considered for re-use. Cables in place should not be shared with other systems, they should be present throughout the facility, they should allow for easy interconnection with other LAN hardware, and proper documentation of the entire wiring system should be available.

Terminating the Cable

An important aspect in the design and operation of the LAN is the location of wiring hubs or termination points. Although a small closet might initially seem appropriate for LAN cable terminations, the ongoing operation of a LAN will underscore the need for a location that provides:

- Security
- Ample floor and wall space
- Adequate electrical power and lighting
- Sufficient conduit space into/out of the room
- Access to a telephone
- Access to external data circuits
- Proper environmental conditions
- Future expansion capabilities

Even though many termination points contain nothing more than connecting blocks for wires, network testing and troubleshooting can be difficult and time consuming if these locations are inaccessible, poorly lit, or otherwise fail to meet the requirements listed above. These issues become very important when a large number of LAN users are unable to access the network due to an outage, and are waiting for technicians to restore operations. In addition to operational considerations, there are planning considerations that support the need for well-designed cable termination spaces. Expansion of the LAN will generally require the addition of more wiring hardware, and the proper design of these wiring areas should allow for the installation of these components.

A wide spectrum of relatively new products exist to allow network reconfiguration, access for testing, and connectivity between network devices and other LANs. These products are installed at cable hub locations as part of a building's cabling system. The simplest of these products is the modular patch panel. The patch panel provides access to each network cable via a telephone-type modular jack. This allows the network administrator to reconfigure, test, connect, or disconnect network devices repeatedly without having to reterminate the cables each time. The individual cable connections represented on the panel can easily be connected to network hardware via modular patch cords.

A newer, more automated version of this device acts as a switching matrix that can be controlled using a data terminal. All network cables and LAN devices are wired to the matrix in order to allow "any to any" device connectivity throughout the building. In addition to avoiding the tangle of patch cords that can result from a large number of patch panels, these matrix switches can automatically provide a record of the specific network connections.

Intelligent Hubs

An even higher level of functionality exists due to the recent popularity of intelligent wiring hubs. Also known as smart hubs or wiring concentrators, these devices can serve as multiplexers, network management control points, matrix switches, bridges, repeaters, terminal servers, and test devices. The intelligent hub often provides an interface between multiple cabling types that exist within a LAN. Many hubs provide for coaxial cable and fiber-optic cable connections in addition to twisted-pair wiring.

Within a network, individual hubs are connected using segments of fiber optic or coaxial cable, while individual network devices connect to the hub via twisted-pair in a star wiring topology. Some hubs offer levels of hardware and cabling media redundancy whereby a cable or component fault needn't cause the LAN to be out of service as repairs are made. The hubs can be networked through the use of modems or bridges to allow a multi-location network to appear as a single, cohesive entity.

As network management control points, these devices provide a window into the operation of a LAN that would otherwise require a multitude of test devices and software packages. Network status, devices, traffic, faults, alarms, maps, and testing procedures can be indicated on a control terminal connected to the hub. These units are likely to become compatible with existing network management packages offered by the major computer systems vendors.

Fiber Backbones

For many existing networks, the interconnection of wiring hub locations is accomplished using coaxial cable, or either shielded or unshielded twisted-pair wiring as a "backbone" connection. The current trend toward higher speed connectivity between intelligent wiring hubs and other network components has contributed to the increased use of fiber optic cable for the backbone. Vendors routinely provide a fiber optic interface on many network devices as well as interfaces to the more conventional cable types. When planning and implementing a new LAN, it is often desireable to include fiber optic cable as a significant portion of the overall wiring system. Fiber promises the higher operating speeds that the networks of the future will require. In addition to greater bandwidth, however, fiber presents a number of advantages over conventional cabling, including:

- Greater transmission distance
- Immunity to electromagnetic interference
- No generation of electromagnetic interference
- Use of less conduit space
- Improved security

The cost of implementing fiber optic cabling has decreased significantly over the past five years as it has become more commonplace, but more importantly, the long-term costs of fiber can prove to be less than that of other cable types. The bandwidth advantages of fiber optics dictate that the long-term ability of fiber to support a network will outweigh that of copper cable. As the copper-based network requires yet more copper cable (and conduits) to support additional capacity and applications, the existing fiber will become increasingly more capable as advances in the devices that connect to the fiber generate even greater bandwidth.

Although many predict that "fiber to the desktop" will become common in the coming years, it is widely used only as a network backbone in LANs being installed today. The absence of requirements for such high bandwidth, the lack of available products at the desktop level, and the cost of providing fiber connectivity to each network user preclude the installation of fiber as a single cabling medium for all segments of a LAN. As plans evolve to cable a facility, the decision to install fiber, even if only for future use, depends largely on the ability to provide future capacity within, and access to, the conduits and wiring pathways of the building.

Managing the Cabling System

Some of the most important aspects of implementing LAN cabling systems are the follow-up activities that occur during and after the installation of the cable. Proper testing, certification, documentation, and ongoing management of the cabling system ensures that the cabling will serve its intended purpose well into the future.

Basic levels of electrical testing should be done as the installation progresses to provide early identification of fundamental flaws in either the products or installation methods. Once the system is completed, a certification that the network can carry the intended data signals should be made. The certification of the system is based on a random sample of wired locations, or by testing every wired location individually. The results of the certification testing should be documented and kept as a part of the overall wiring system documentation. One of the advantages of having a complete, well documented certification test is that it provides a benchmark by which future system testing can be compared. The investment of time and money in certification testing can be recovered many times over during the life of the cabling system. Comparing future test results against the initial certification measurements helps avoid extensive and time consuming troubleshooting and recabling. Documentation indicating the existence and capacity of cables within the network wiring system should be checked against the initial installation and, most importantly, be kept up to date.

Unfortunately, it is common for a network administrator or technician to become apathetic toward updating the cabling documentation when a new cable is added or a change of any kind is made to the wiring system. The modification of blueprints or cable records to reflect these changes is often seen as a minor responsibility in the context of the more technically demanding aspects of an administrator's function, and updates are often overlooked. Especially in a large network, the deterioration of cabling records can come back to haunt those that work with the wiring. Taking the time to maintain the cabling system documentation on an ongoing basis will always prove more efficient than trying to figure out the network during a system outage.

Trends

The incredible speed at which innovations in communications technology materialize can create headaches for LAN implementors. When planning a network to meet the requirements of every eventuality, it soon becomes evident that the breadth of future services that might be delivered over the LAN is difficult to anticipate. The emergence of Integrated Services Digital Network (ISDN), Fiber Distributed Data Interface (FDDI), and high-bandwidth applications such as image processing can make an appropriate cabling strategy seem to be a moving target.

ISDN

The availability of ISDN services, although rolling out at a slower than expected pace, is expected to increase the level of LAN-to-LAN communications due to the high-speed switched data connections that ISDN can provide. Basic and primary rate ISDN services at speeds up to 1.544M bps are now, and will increasingly be, utilized extensively in connecting LANs via the public telephone networks. The introduction of broadband ISDN services at data rates up to 140M bps, or multiples thereof, are unlikely to occur until the late 1990s.

This report was prepared exclusively for Datapro by Bernard A. O'Donnell, Director of Communications for the Office of Information Technology of the State of Connecticut. Mr. O'Donnell has over 15 years of experience as a consultant and manager in voice and data communications. He has previously held positions with the U.S. Air Force, Milton Bradley Co., United Technologies, and New England Microwave Corp., his own consulting firm.

FDDI

The FDDI specifications for a 100M bps, fiber optic based local area network architecture have been embraced by a number of LAN vendors. FDDI networks offer LAN users an increased level of bandwidth that can meet the demands of LANs interconnected within an enterprise-wide distributed processing environment.

Standards Compliance

Despite the tumultuous changes in the networking environment, those implementing cabling systems have found that a standards-based approach will serve them well into the foreseeable future. As more organizations implement cabling based on the EIA/TIA standards, the market for products will be largest for those vendors addressing standard cabling methods. Users are intensifying their demands for standards-based products throughout the information technology arena. The response and involvement of manufacturers in participating in the development of these standards is heartening and unlike any past efforts.

One indication of the efforts of manufacturers in adapting to existing standards is the current debate to adopt a standard to allow FDDI signals to operate over either shielded or unshielded twisted-pair cable. Groups of vendors have rallied around each type of cable, largely based on the architecture of their own existing products. Much of the impetus for the use of copper cable for FDDI is based on the fact that users have so much copper already installed. If this debate had taken place ten years ago, each vendor group would likely have adopted the cable type that best suited its own needs, and no attempt would have been made to develop a single standard. Furthermore, it would have been unlikely that vendors would have bothered to deviate from the fiber-based FDDI implementation in the first place. Although all three options for FDDI may eventually be marketed, the fact that these parties are working closely together in the standards crusade is beneficial for the end user.

The prevalent wiring systems offered by the major computer and communications vendors generally adhere to the EIA/TIA standards. These standards accommodate existing networking topologies and data transmission rates, and provide users with a means to migrate to ISDN and FDDI-based services without recabling their facilities. The end user who insists on a standards-compliant cabling plan will find the optimum level of multivendor, multi-service capability that can reasonably be expected from a wiring system. ■

Trends in Intelligent Wiring Hubs

Datapro Summary

LAN hubs weren't always as functional as they are now. They started out as unintelligent mechanisms designed to solve wiring problems. State-of-the-art hub technology now includes network management software based on the Simple Network Management Protocol (SNMP). Intelligent wiring hubs permit LAN managers to monitor the network from a single vantage point. As LAN managers move toward automated network management, intelligent wiring hubs play an increasing role in the automation process.

There ought to be an IQ test for new intelligent wiring hubs. The advantage that smart wiring hubs have over traditional hubs is that they allow users to manage the network from one place.

But users say intelligent hubs should be even smarter. The hubs on managers' wish lists would have very simple command interfaces and would be able to self-map the configurations they administer.

It is good to remember, however, that hubs began as unintelligent devices, simple tools to eliminate wiring snarls. And although they continue to solve local network wiring problems, hubs have evolved to the point where they now also play host to internetwork devices such as bridges and routers.

Network Management

The most sophisticated intelligent hubs save network managers steps by including net management software, often involving the Simple Network Management Protocol (SNMP). They monitor and report on such network activities as traffic load, error conditions, and broken cables.

Even more sophisticated wiring hubs can monitor LANs down to the port level. They can also enable and disable ports remotely so that managers can configure networks from their desktops.

Increasingly, intelligent wiring hubs are capable of working with different LAN technologies, including Ethernet, token ring, and Fiber Distributed Data Interface. They cannot, of course, enable transmission of data between LAN types. That is, not unless they have built-in bridging or routing functions.

This is coming along, too. Intelligent hub manufacturers are building bridging and routing capabilities into their products, allowing network managers to connect various kinds of LANs.

Finally, some intelligent hubs are gaining sophisticated security features, and others boast redundant modules to bring a measure of fault tolerance to the network.

One major advantage of intelligent wiring hubs is that they allow managers to take charge of the LAN from a single point.

With traditional wiring hubs, a manager would have to walk to the closet containing the concentrator suspected of causing the problem and unplug desktop devices until the faulty connection was located. The wiring closet could be several flights—or several buildings—away from the manager's office.

This Datapro report is a reprint of "A Guide to Wiring Hubs: Smart, Smarter, Smartest" by Kimberly Patch, pp. 1, 36-37, 39, 44, and 46, *Network World*, March 11, 1991, Volume 8, Number 10.

The advantages of intelligent wiring hubs grow when network management capabilities are added to them. When using a hub that has SNMP capabilities, for example, a net manager can see the onset of failures, having been alerted by preset alarms.

Intelligent wiring hubs bring the network manager "the ability to collect statistics [such as] error rates and the number of collisions," says John McConnell, president of Interworking, Inc., a market research firm in Boulder, Colo. They can "disable or enable certain ports and take misbehaving segments out of the LAN."

In addition, intelligent wiring hubs enable users to create logical networks.

"You are star-wired from the hub; it gives tremendous network advantages," says Michael Howard, president of Infonetics Research Institute, Inc., a market research firm in San Jose, Calif. For instance, within some intelligent hubs, system managers can create logical networks and bridge networks. They may also be able to attach routers to the hub for connecting to LANs or wide-area networks.

Pyropower Corp., a San Diego-based company that manufactures large boilers for power plants, recently built a new headquarters and took advantage of the event to set up a sophisticated network that uses unshielded twisted-pair wiring and modular intelligent wiring hubs. The setup is a world away from Pyropower's old hub network.

With the old hubs, managing the network was "like a Christmas tree light scenario," says Klay Elwood, manager of MIS for Pyropower. "Something would happen and you'd give somebody a terminator, and they'd run from station to station plugging in until they found the station that was down. Sometimes they'd never find it, and we'd down the whole network and bring it up again and disturb everyone."

The addition of intelligent hubs supporting SNMP allows Elwood to "manage [the whole network] from a workstation and provide the type of design redundancies and management capabilities that will allow us to serve our customers without significant interruptions," he says.

Bridges and Routers

In addition to network management, better interfaces and the support of various media, many of today's intelligent hubs are gaining the ability to support bridges and routers.

Partnerships are forming between hub vendors and LAN interconnect suppliers to bring bridge and router functions to intelligent wiring hubs.

In 1990, for instance, hub vendors Cabletron Systems, Inc., Chipcom Corp. and SynOptics Communications. Inc. all signed agreements with router vendor Cisco Systems, Inc. for Cisco to build router cards that will slide into slots in their respective intelligent hub offerings.

Hub maker Ungerman-Bass, Inc. last December signed a technical development and marketing agreement to use advanced Computer Communications multiprotocol bridge/router technology in its hubs.

SynOptics is planning to announce remote routers as well as bridges and routers between token ring, Ethernet and FDDI later this year, according to Steven Moustakas, the company's director of product management.

Another trend is the ability to concentrate and monitor terminals within a hub.

For instance, Banyan Systems, Inc. introduced a line of terminal servers last summer that plug directly into Cabletron's, Chipcom's and SynOptics' intelligent hubs.

In addition, SynOptics last October announced a technology exchange agreement with Xyplex, Inc. to incorporate a terminal server made by Xyplex into the SynOptics hub.

Intelligent Wiring Hubs

There are two main types of intelligent wiring hubs. The most intelligent of the two are the large, self-contained hubs sold by Cabletron, Chipcom, SynOptics, Ungerman-Bass and Fibermux Corp. These hubs support high-level capabilities such as multiple media and access methods, routing and bridging, and monitoring down to the port level. They also allow remote enabling and disabling of ports.

Other self-contained hubs, such as David Systems, Inc.'s David ExpressNet Hub, which supports only Ethernet, are less talented and less expensive.

AT&T, Hewlett-Packard Co., Proteon, Inc. and others have taken the modular rack'em, stack'em approach. This kind of hub tends to be less expensive than the all-encompassing hub but is generally less sophisticated. Modular hubs usually support several media and offer good network management capabilities but support only one access method.

Today's large self-contained hubs contain slots into which cards plug. Typically, each of the cards, sold separately, supports eight, 12 or 16 users for many as 132 desktop devices supported per hub.

A user can mix and match these cards, installing three Ethernet cards one token-ring card, an FDDI LAN and a terminal concentrator, for instance. The trends in this type of configuration are larger shells that support more users and a wider variety of cards.

The rack'em, stack'em variety comes in modules that support eight, 12 or 16 users each and can be rack mounted or stacked.

These external modules can be connected with thin-wire Ethernet and managed as if they were part of a larger chassis.

The larger shells tend to be more expensive per port costing around $50,000 for a 132-port system, or between $150 and $600 per port, depending on how many of the slots are used. The rack'em, stack'em variety is less expensive and runs closer to $150 to $200 per port.

A user that wants to build a network slowly and does not have a lot of cash to invest in equipment that will not be used until later might want to take the modular approach. Users with large networks that need the most sophisticated network management software would do well to look into the self-contained hub approach.

"It's analogous to buying an all-in-one stereo system vs. buying a preamp, CD player, power amp and cassette player, and stacking them up and connecting them in the back with wires," says Infonetics Research Institute's Howard. "The difference is that the one big box is nicer."

Cabletron, for instance, offers its Multi-Media Access Center (MMAC) in three sizes. The latest offering is the MMAC 5FNB, which supports as many as 120 ports in up to five Ethernet, token-ring and FDDI networks and includes a management/repeater module. The company's other MMAC models support three and eight networks each.

MMAC can be configured with a second power supply for increased reliability. Cabletron offers two network management software packages. Users can manage the net from any site using the company's Spectrum software.

Cabletron's remote LanView package allows users to manage the MMAC through a modem.

Another example of the one-box intelligent hub is Chipcom's 17-slot ONline System Concentrator. It is a fault-tolerant hub that supports any combination of three Ethernet, token-ring or FDDI networks.

Modules available for the slots are the eight-port Twisted Pair Ethernet Module and Transceiver; the four-port Fiber Module; the Ethernet bridge module, which takes two slots; and the Ethernet Network Management Module.

The Network Management Module is controlled via a terminal connected to an RS-232 port on the module. The company claims the external Midnight Bridge module can also be controlled via the Network Management Module.

This spring, Chipcom is scheduled to announce Network Control System software that will allow users to control the hub from any desktop connected to it.

Subsequent introductions will include internal bridge and router modules, which will each take up two slots in the chassis.

SynOptics' System 3000, Ungermann-Bass' Access/One and Fibermux's CrossBow complete the list of large self-contained smart hubs that support Ethernet, token-ring and FDDI while also letting the user monitor down to the port level and enable and disable ports remotely.

AT&T, which uses the rack'em, stack'em approach, earlier this year announced its 13-port Starlan 10 Network Smart-Hub. AT&T's Starlan 10 Network SmartHub Manager supports SNMP, can draw topological maps down to the end user and includes intrusion control and eavesdropping protection security features.

The Starlan 10 Network SmartHub Manager converts an Ethernet broadcast into a logical point-to-point transmission path to gain increased security. It identifies a data packet's intended destination by matching its destination address with the SmartHub's authorized node addresses.

In addition, it prevents unauthorized nodes from transmitting over the network by matching source addresses.

AT&T also offers the Starwan Multi-Bridge module, which can interconnect two FDDI networks, four Ethernet networks or four Ethernets and one FDDI network.

HP's EtherTwist products are also rack'em, stack'em systems that include l0BaseT-compliant hub modules supporting from 12 to 48 ports each; an eight-port fiber-optic hub; local and remote bridge modules; eight- to 32-port adapter cards; OpenView Bridge Manager; and OpenView Hub Manager software.

HP's OpenView is based on the X Window system, supports SNMP and allows managers to monitor hubs or bridges from one dedicated personal computer on the network.

The company plans to announce before year end a router that will fit into its smart rack and be managed by OpenView, according to Alan Housley, HP's product marketing manager.

In contrast with most hub makers, which have tackled multiple LAN types, HP has no plans to add token-ring modules to the EtherTwist family, Housley says. "Our plans are not focused on token ring because there are only a small amount of users who want both [types of LAN]," he asserts.

A similar one-LAN specialist is Proteon, whose product only works with token ring. Proteon has no plans to add other media to its modular family of hubs, according to Mary McGregor, Proteon's public relations manager.

The company sells the Series 70 intelligent Wire Center and provides an SNMP-based net management system called Overview.

Not Smart Enough

Users, of course, have their own views about whether today's intelligent hubs are truly smart. Many of them think hubs need to be a whole lot smarter.

The biggest complaint users have about today's intelligent wiring hubs pertains to the user interface. Network management user interfaces are getting better, "but they're still in their infancy," says Pyropower's Elwood.

Some users are asking for easier to handle command interfaces and a simpler way of creating net maps of LAN wiring systems.

Vendors say both of these capabilities are improving. Interfaces are becoming more like the graphical user interfaces of Apple Computer, Inc.'s Macintosh, and some firms' products can automate the LAN map. But more effective use could still be made of today's graphics capabilities, users say.

In fact, according to one user who requested anonymity, network interface menus are poorly designed. "If I just want stats on one port, I have to go through about three menus." he says.

Future Hubs

The attractiveness of the new hubs is helping spark an interest in net management standards. Without such standards, the choice of one hub means that all other hubs in the network must be supplied by the same firm.

According to Infonetics Research Institute's Howard, the wave of networking in the future will be more connections and more collected statistics.

"The most flexible box would have Ethernet and token ring at both speeds and FDDI [with] different media options," he says. "It would have local bridging and router connections to local or remote networks, terminal server connections and SNMP network management."

Hubs will increasingly connect more elements of the network and become an increasingly important part of the network, says Interworking's McConnell.

Future hubs are also likely to be physically smaller than current ones, according to hub vendors. AT&T is working on very large-scale integration chips for smart hubs, says Bob Donnelly, an AT&T district marketing manager.

But despite recent networking advances, promises for the future and network managers' wish lists, many users still spend more time putting out fires on the network than using the reporting capabilities of smart hubs.

According to the aforementioned anonymous user who recently switched to modular intelligent hubs, the primary advantage of these hubs is that they "give you more control over problem situations."

Although his intelligent hubs support SNMP, the user says he hasn't had time to look into using that capability. "We [still] put out fires most of the time," he adds.

Nevertheless, users say they are moving toward more automated network management and intelligent hubs are a major part of this trend.

And even just as instruments to deal with crisis situations on the net, intelligent wiring hubs have proven useful to managers. Now they can put out their fires while sitting down. ■

The Next Generation Hub

In this report:

Datapro Summary

Data networking continues to grow and change. Network planners need to understand today's market needs and anticipate future needs. This report provides information on the next generation hub and its potential for simplifying network capabilities by incorporating a hierarchical network management architecture.

Rapid change in the data networking arena means that the creation of products for the next generation is a lot more than a problem in system design. Successful products, which respond to real market needs, must often be designed before these needs are fully known or understood. This puts the system designer in a position where he or she must intelligently evaluate current conditions to correctly anticipate prospective needs.

This report considers the market requirements for the next generation hub and outlines an architectural approach that will fulfill future needs. The major objective of this architecture is to simplify the data network. The approach described in this report offers in a single box a full range of standards-based data networking functions currently provided by a variety of systems, including repeaters, bridges, and routers. The new modular architecture is open and easily extended and offers maximum flexibility in adapting the system to specific user requirements. Moreover, it simplifies the data network by incorporating a hierarchical network management architecture, which delegates network management functions to the lowest possible level.

This Datapro report is a reprint of "The Next Generation Hub" by Dr. Dono van-Mierop, pp. 31-37, from *3Tech*, Volume 2, Number 1, Summer 1991. Copyright © 1991 by 3Com Corporation. Reprinted with permission.

This report is written for network planners who require information on the next generation hub and who need to simplify their network's management capabilities.

Market Demands and the Next Generation of Data Networks

Why did the introduction of the PC and the development of LANs spark explosive growth in the data networking marketplace? LANs showed users the power of sharing information and distributing applications in terms of real, bottom-line benefits to their companies. The logical way to obtain additional benefits was to create systems that would permit even more of this sharing, i.e., to connect the LANs together.

There are three principal reasons for the continuing growth of data networks. First, PCs and LANs are still increasing in all types of organizational environments. Second, increasingly more computations are performed by distributed processing. This "machine/machine" interaction requires more communication bandwidth. And, finally, new 1/0-intensive applications (such as interactive graphics and image processing) have enormous appetites for data storage and transmission capabilities.

In tomorrow's business environments, therefore, data communications services will be as pervasive as today's telephone services. The data network will be viewed as a "utility," with the end user depending heavily on the service and its quality. In most environments, the penalty for poor service is high. For example, a delay in placing a trading order in the stock market due to an outage in the data network may cause millions of dollars in losses. This represents substantially more than the cost of the equipment itself. In addition, users need simple access to the data networking service, as simple as picking up the phone and hearing a dial tone.

Today's users already expect total availability and easy accessibility. Tomorrow's users will also be eager to employ applications that place new levels of burden on the networks and their managers. From the end user's perspective, next-generation data networks will have to be:

- Standards-based at the physical connection level. The end user primarily cares about a transparent interconnection of Ethernet, Token-Ring, and fiber distributed data interface (FDDI) networks. The data networking service should be independent of things like higher layer protocols.
- Reliable. Business is starting to depend on the reliability of data networks. Therefore, many users require 100% availability. The next-generation data network will have to include the mechanisms and tools to ensure that level of reliability.
- Flexible. Users will want a full range of possible tradeoffs between cost, performance, and level of service. They will want systems that are scaleable and that adapt easily to change in ways that make network interconnection simpler, rather than more complex.

While the end user views the data network from the LAN perspective, the network managers responsible for planning, installing, and maintaining the data network have a global perspective. These managers require:

- Flexibility in installing the networking equipment in a physical cable plant, possibly using existing wiring
- Management tools for predicting errors (and for quickly and easily detecting and correcting them), as well as for other management tasks, such as network configuration, alteration, and performance optimization
- Cost effective equipment and installation procedures, since the cost of down time may he higher than the equipment itself in critical applications

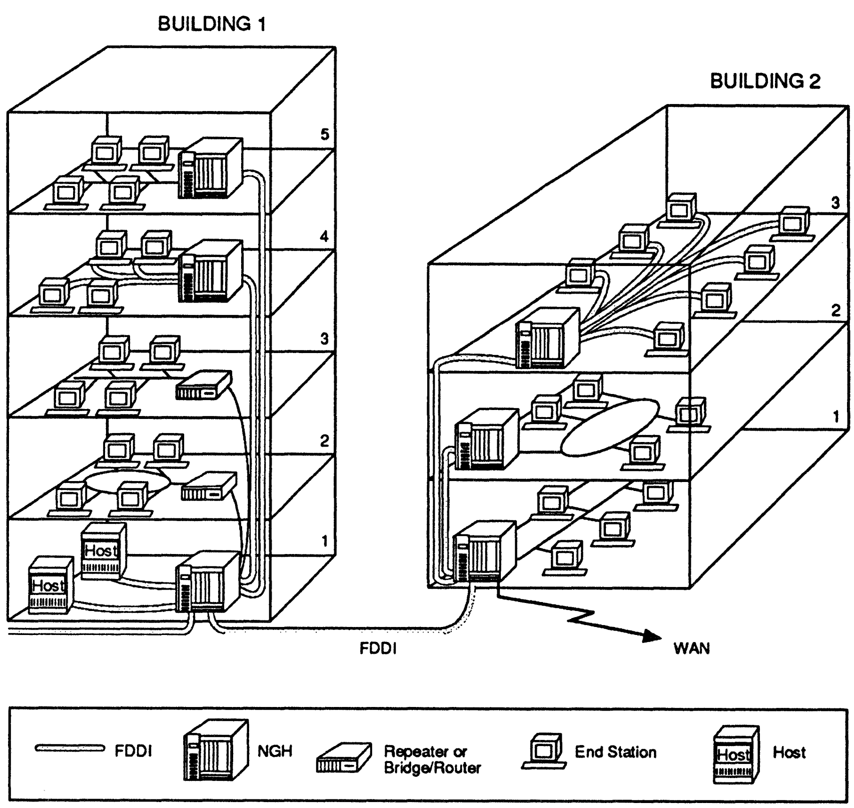
Specifying the Next Generation Hub

The increased demand placed on data networks will require solutions based on new architectures. This report describes the next generation hub (NGH) as the basis for such an architecture. Before discussing the NGH, we will offer a model of the future environment.

It is helpful to view the data network as a three-tiered model. LANs, campus area networks, and WANs are related to topological entities. For example, LANs are typically single physical segments of Ethernet, Token-Ring, or FDDI networks on a floor. In larger networks, LANs may contain multiple physical segments comprising a single, logical LAN (such as bridged LAN segments in a single building). LANs are typically tree structured with roots in the wiring closet.

Hubs in the wiring closets in one or more buildings on a campus are typically interconnected with a campus backbone. Current generation hubs are limited to LAN-repeating functions. There is a trend toward using FDDI as the backbone, thus dramatically increasing the communication capacity to the wiring closet. This opens up an opportunity to increase the functionality of the hub. NGH is

Figure 1.
Typical Deployment of New Generation Hubs



a hub that includes not only a repeating function but also bridging, routing, and interfaces to various networks and network management.

Figure 1 shows possible locations of the NGH within the campus: in the communication room or in the floor-wiring closet. The NGH-based data networking architecture is simple, because it serves as a universal building block. The NGH contains LAN functions for end user connections, interconnects LANs on a campus, and provides WAN interfaces—all in the same type of box.

For an NGH to succeed, it must be able to support networks that are standards-based, reliable, flexible, and manageable. These networking goals can be attained by designing an NGH that:

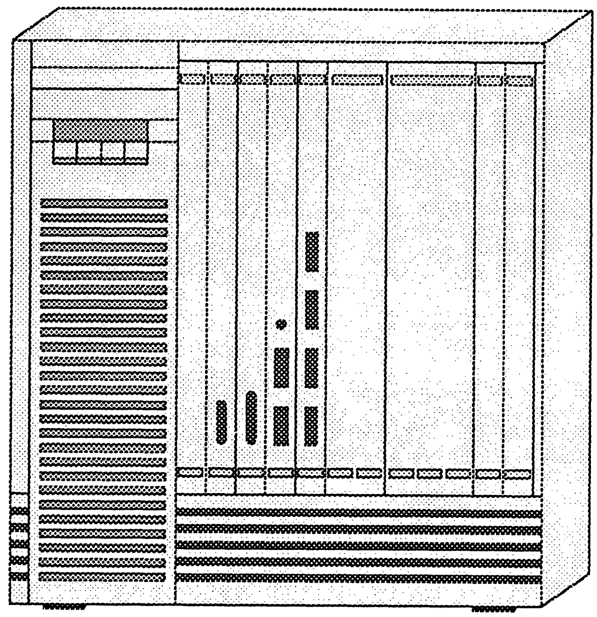
- Provides a total solution, satisfying all user network service needs. The NGH must support any combination of data networking services, including repeating, bridging, and routing functionality, as well as both local and remote connectivity. It should also provide interfaces to Ethernet, Token-Ring, and FDDI end users.
- Pushes reliability up to the 100% availability limit. This means a simple architecture, no single point of failure, and sophisticated fault prediction and recovery mechanisms.
- Has an extendable architecture. The NGH should be capable of adding functionality on demand by users plugging in the appropriate boards or loading the necessary software. Moreover, the NGH is the place to ensure flexibility and scaleability through a range of price, performance, and service options.
- Never creates traffic bottlenecks. To handle substantially increased traffic, the NGH will require a much faster throughput rate than that of today's hub products. In addition, there should be a simple method for adding capacity.
- Protects investments by ensuring compatibility with users' existing equipment. Since the NGH should be completely interoperable with all standards-based equipment, it will have to support both transmission control protocol/internet protocol (TCP/IP) and open systems interconnection (OSI), as well as other popular standards, such as Xerox Network Systems (XNS) and DECnet.

Of course, fulfilling these requirements with today's technology would require several devices. But the NGH should fulfill them as a single-box solution. The reason for this is straightforward. It is inherently easier to manage a single device than several devices. This is the primary reason for requiring that the NGH provide a total data networking solution.

NGH System Architecture: A Modular Approach

The requirement for both an extendable architecture and a total solution immediately suggests a modular strategy for the NGH system architecture. A modular architecture would allow the NGH to be extended to the physical limits imposed by the size of the equipment rack. Moreover, it would provide total flexibility. Functional modules could be added to, or removed from, the system based on users' required data networking services. The probable physical appearance of the NGH is depicted in Figure 2.

Figure 2.
A Typical NGH Enclosure



The modular approach is best described in terms of two types of elements: basic building blocks and extension modules. Basic building blocks, which provide the framework for NGH functionality, include the backplane, buses, and system manager. The extension modules provide required data networking functionality: interfacing, repeating, bridging, and routing. These architectural components are illustrated in the schematic model in Figure 3.

The key architectural concept that serves as a basis for flexibility and potential high performance is distributed processing combined with network management functions. This distributed processing power means that performance will not degrade, even when more connections are added. The performance and cost can be tailored to individual needs.

In contrast with the distributed modules, network management functionality is inherently centralized. The system manager is the central element that reconciles any differences among the distributed modules.

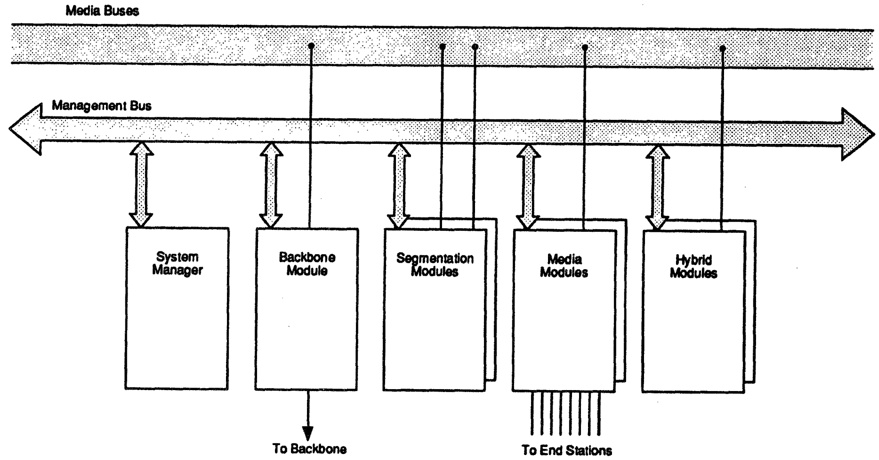
Basic Building Blocks

The basic building blocks provide a framework for the full range of network services on a single platform. These basic building blocks include media buses, a high-speed bus, a system management bus, the system manager, and the power distribution system.

While current generation hubs include only one network bus per type, the NGH will provide multiple media buses for each transmission medium, such as Ethernet, Token Ring, or FIDDI. These buses will integrate the LANs into the NGH, providing users with the flexibility of tailoring the number of connections to their current or future needs by adding modules. Multiple buses of the same type allow the integration of several LAN segments.

Because of the need to support very high data rates, the backplane will probably include a high-performance bus or switch for fast data transfer between the modules. This bus

Figure 3.
NGH's Modular Architecture



must ensure that there are no bottlenecks in intermodular communications. The backplane contains several buses. For example, using multiple FDDIs for this purpose has the advantage that it could be extended out of the box, thus reducing the number of store-and-forward hops required between any two end stations. Multiple FDDI buses not only provide both high performance, and also redundancy that enhances reliability and is very instrumental in fault isolation and bypass.

Comprised of its own high-performance processor, memory, and peripheral support, the system manager communicates with each module, collects appropriate statistics and other information, monitors and tracks performance, and places modules off-line in the event of failure. The cornerstone for the hierarchical network management architecture, the system manager provides centralized control over the distributed extension modules.

The system management bus is a general-purpose bus (for example, VMEbus). It must provide the bandwidth for the system manager to communicate effectively with all extension modules; yet, its interface logic must be inexpensive, so that it does not increase the cost of the individual modules.

Finally, the power distribution should include redundant power supplies. For maximum reliability, it must also support "hot swapping," which is the ability to add or replace modules without powering down the entire system. For simplicity, every slot should have enough power to accommodate any module.

Key Extension Modules

System elements, referred to as extension modules, fit into the framework provided by the basic building blocks. These modules, which provide the actual data networking services that are the main reason for building the NGH, include:

- Media modules. These modules function similarly to today's hubs. They provide high-performance communications between all connected devices to form a logically unified network. They do not, however, provide any filtering but forward all data one frame at a time. Media modules provide connections to end stations, using one of the standard LAN protocols.
- Segmentation modules. These modules provide bridging and routing services, permitting the same degree of network segmentation possible with today's bridges and routers. Integrating these functions into a single hub increases its reliability by eliminating cables and by allowing better network management. Segmentation modules are critical for performance and reliability, since they isolate LAN segments that operate concurrently and with limited interaction.
- Backbone connectivity modules. These provide interfaces to the campus or WAN backbones. Campus interfaces are typically FDDI and could be Ethernet. WAN interfaces will probably support connection to a variety of services including, T1, T3, Sonet, X.25, Frame Relay, or switched multimegabit data services (SMDS).
- Hybrid modules. These modules provide mix-and-match functionality, such as WAN routing services or a combination of repeating with bridging/routing. The modular architecture permits virtually unlimited functionality in such hybrid modules.

Gaining Simplicity Through a Hierarchical Approach to Network Management

The best thing about developing the NGH from scratch is that we can build network management right into it, rather than adding it later as many vendors have done with the current generation of products. This allows us to determine the optimal method for incorporating network management functionality.

The modular architecture of the NGH makes it possible to implement a hierarchical approach to network management. In this approach, components of the network management are implemented at different levels. Each network management problem is solved at the "lowest" possible level. Network management functions are delegated to lower levels if possible. This ensures the fastest possible fault recovery, while minimizing the computation and communication resources needed for the solution.

The media modules contain the lowest level of network management. These modules implement the physical protocol in hardware and include capabilities like enabling/disabling ports. In FDDI, for example, the media module performs a sophisticated link establishment protocol autonomously. Although these modules typically do not include a processor, they can be directly accessed by the system manager for configuration or status collection.

Extension modules that handle higher layer protocols perform a variety of network management functions in their hardware and software. They would typically expose a management information base (MIB) to the higher level network management.

As its name implies, the basic building block referred to as the "system manager" provides the next (central) level of network management functionality for the NGH. The system manager is responsible for collecting network management data and communicating with the network management station.

The system manager can perform many of the same functions as today's network management stations or even human network managers. For example, counters in the media modules need to be polled at relatively high frequency to provide responsive trend analysis. Rather than having the polling activity load the communication path to the network manager and its processor, the system manager performs the data gathering and trend analysis within the NGH, alerting the network manager if it detects abnormal behavior.

Many users find that the reliability of their networking hardware is fairly high. However, once a fault occurs, it typically disappears quickly, with no clue as to what caused it. The system manager is the ideal level to continuously monitor the network and record the history. Once a fault occurs, this historic perspective should provide higher level management with an audit trail to the problem.

Any problems that the system manager cannot solve would be escalated to the network management station. And, presumably, (human) network managers would be notified about any problems that cannot be solved by the network management station. By delegating management functions to different levels within the hierarchy, this network management architecture is clearly a major contribution to the simplicity of the data network.

Conclusions

NGH will be a single device offering a full range of data networking services and unprecedented flexibility, combined with excellent price performance. To ensure that it attains such a high standard, this system will probably be implemented through a modular, extendable architecture that includes a great degree of fault tolerance. These devices will become the central focal point for the next generation of networks. These future data networks will offer a level of data communications services comparable only to the services of today's telephone system: total availability combined with simplicity of use. ■

Wireless LANs

In this report:

Datapro Summary

Eliminating costly and space-consuming cabling from their premises is a priority for many users. With the local area network (LAN) market booming, many vendors are introducing wireless LANs, eliminating the need for coaxial, twisted-pair, and fiber optic LAN cabling. Varied technologies are being used to implement wireless LANs, including infrared light and spread spectrum (UHF radio). As with any new technology, however, there are benefits and limitations inherent in wireless LAN technology. Still, there are several wireless LAN products currently available from vendors including BICC, Motorola, and NCR.

Wireless Technologies

The basis of what is traditionally referred to as wireless LAN technology is radio. Radio uses a broad portion of the lower end of the electromagnetic spectrum as a vehicle to encode and ship information from one place to another. The electromagnetic spectrum is made up of energy which includes heat (infrared), light, X rays, and radio, all of which travel invisibly at the speed of light as waves. The characteristics of the waves are physical length (measured in meters) and the frequency at which they oscillate (measured in hertz, or cycles per second). These two attributes determine the way in which waves behave. Actually, there is an inverse correlation between wavelength and frequency; as wavelength increases, frequency decreases.

As the frequency of waves increases, the electromagnetic radiation becomes known first as infrared, then light, then ultraviolet radiation, then X rays. The increase in wave frequency takes us across the electromagnetic spectrum (see Figure 1).

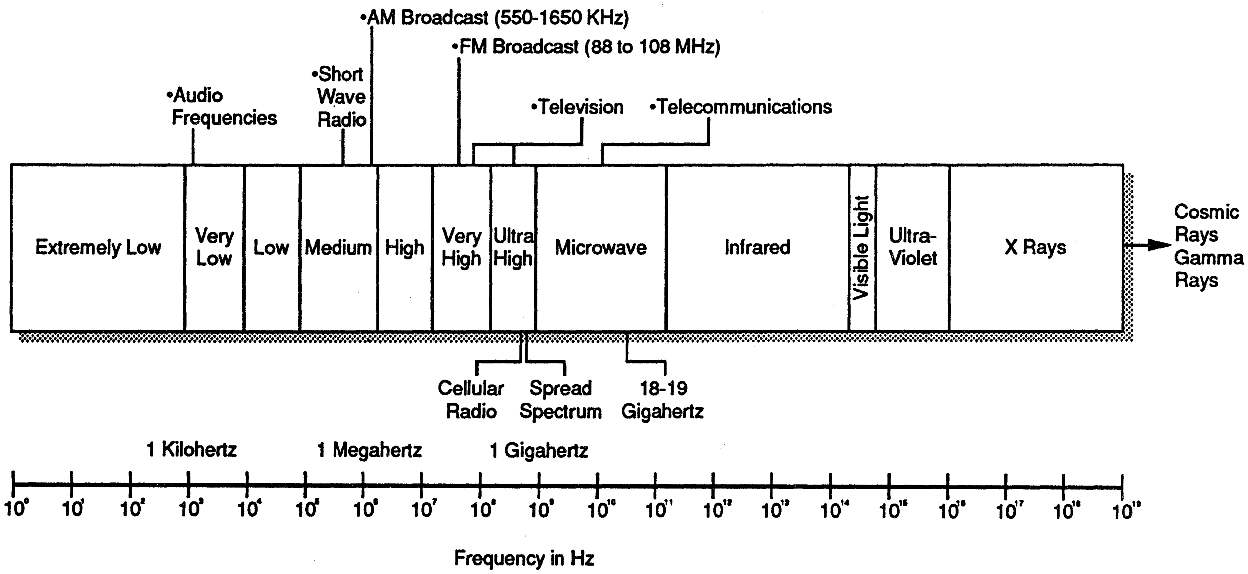
The strength of radio waves is a function of amplitude, which is the strength of their oscillation and not their frequency. Radio uses a carrier frequency—a certain fixed, constant frequency—where it encodes an information signal by modulating or varying the amplitude (AM, or amplitude modulation) or frequency (FM, or frequency modulation) of the carrier frequency in direct proportion to the signal. The encoded radio frequency signal is radiated into the air, after it has been amplified, through an antenna. Once the signal hits the antenna of the receiving radio, it converts it back into electrical form. The receiving radio then takes the information signal from the carrier frequency and separates and reconstructs it through a process called demodulation. The signal is then amplified, and the decoded information is sent to the output device (most commonly a speaker).

Information can be carried in analog (natural waveforms of information) and digital (information in discrete, discontinuous, usually binary bit) form.

Wavelengths vary considerably and can react very differently to different items in the environment—they can bounce off rocks, be absorbed by wood, and even react differently to changes in atmospheric pressures. All radio frequencies exhibit their own characteristics. For instance, low-frequency, long-wavelength radio frequencies (RFs) can pass through earth, water, and even human-made structures.

—By Bernard J. David
General Information Services, Inc.

Figure 1.
The Electromagnetic Spectrum



The electromagnetic spectrum is made up of energy which includes heat (infrared), light, X rays, and radio, all of which travel invisibly at the speed of light as waves. The characteristics of the waves are physical length (measured in meters) and the frequency at which they oscillate (measured in hertz, or cycles per second).

Source: Motorola, Inc.

UHF Radio Systems or Spread-Spectrum Technology

The first wireless devices used the ultrahigh frequency component of the electromagnetic spectrum. This technology became known as spread-spectrum.

The essence of spread-spectrum technology is that it is spread across a range of radio frequencies instead of being sent on one specific frequency. The result of this spreading is a unique "waveprint" which can only be intercepted by a device that understands the spreading which has taken place by the sending device; data is therefore difficult to intercept. The spreading effect in spread-spectrum technology leaves this option with improved security—no one frequency can be readily tapped to siphon data.

The technology was developed during World War II for military applications which were based on antieavesdropping and antijamming properties. In 1985 the Federal Communications Commission (FCC) set aside selected bandwidths for commercial data transmission by radio waves. These three bands of frequencies are 902MHz to 928MHz, 2400MHz to 2483.5MHz, and 5725MHz to 5850MHz. In setting these frequencies aside, the FCC limited commercial installations to a low power level of one watt, which results in a transmission that can only travel 800 feet.

These bandwidths are essentially the upper end of the UHF band. The upper end of the UHF band is not only used for spread-spectrum radio technology but also for television, FM radio, and cellular radio. They all use several small, narrow bands, within a general region of the spread-spectrum bands, as carrier frequencies. As a consequence, there really is not a lot of available bandwidth for spread-spectrum technology.

Spread-spectrum technology (also called SST) operates on a non-exclusive spectrum of radio frequencies. There are other radio signals being sent across these frequencies, and the signals are therefore susceptible to interference. The maximum speed or signaling rate which the data can travel is 230K bps.

Wireless In-Building Networks (WINs)

At the highest level of UHF frequencies, frequencies exist within the electromagnetic spectrum which are virtually interference free. Spectrum reuse is therefore more easily done here, than at a lower, more crowded frequency.

These frequencies (18GHz to 19GHz) are in essence a blend of UHF frequencies and infrared characteristics. At these higher bandwidths, transmission speed is much quicker. Presently, transmission speeds of 15M bps are being achieved while 80M to 100M bps are attainable. Due to the uncrowded nature of these bandwidths, there is enough bandwidth available to handle these higher speeds that will be found in future computer systems.

This higher bandwidth is to be licensed by the FCC. The FCC will also assign frequencies on which signals may travel. The FCC now calls the 18GHz to 19GHz frequencies the Digital Termination Service (DTS) band. Motorola coined the name Wireless In-Building Networks (WINs) as the vendor applied it to this bandwidth. One problem with this frequency is that signals are often blocked by large structures. While it is a high enough frequency that it does not interfere with other electronic equipment, signals can only pass through drywall surfaces. Motorola has developed a lot of technology around the wireless use of this DTS band. It includes the following.

Six-Sector Intelligent Antenna: This antenna is set up to continuously select the best signal for each individual data transmission.

High-Performance RF Digital-Signal Processor: This processor is a radio frequency digital-signal processor which is used for data synthesis and recovery. Motorola embedded this technology in CMOS technology, which incorporates specialized modulation and demodulation techniques with a bit error rate of 10^{-8}.

GaAs MMIC Module: Gallium Arsenide Monolithic Microwave Integrated Circuit (GaAS MMIC) technology enables the DTS technology to be miniaturized so that a system module is the size of a deck of cards. This module contains five GaAs integrated circuits which both transmit and receive the 18GHz RF energy.

Packet Switch and Network Interface on a Chip: This chip has a three-level switching architecture which is intended to manage, organize, route, and ensure the integrity of the high-speed datastream. One million connections per second can be processed through its on-chip switching capability.

Infrared

Infrared wireless LANs operate at an even higher frequency on the electromagnetic spectrum. Infrared is immune to electrical interferences and is quite available in terms of various frequencies. The technology is line-of-sight (even though it is invisible to the human eye) in that it cannot penetrate dense surfaces. However, it can be reflected off some common point for linking transmissions together. The speed of infrared transmissions is currently 1M bps, but in the future it is anticipated such transmissions could travel at 10M bps through products on the market.

Carrier Current Radio

Wireless alternatives must include a mention of carrier current radio. The basis of this technology is to send low-frequency radio signals over AC power lines. The technology is inexpensive but constrained to a narrow bandwidth. It has the same distance constraints as does RS-232 cabling. Its maximum signaling rate is 115.2K bps.

Why Use Wireless LANs?

Why should you use wireless local area networks? There are a host of reasons. The most commonly cited ones are:

Inflexibility of Cabling: When cable has been laid in a building, it is often difficult and expensive to change the configuration of that cabling—walls have been closed off and they must be opened again to rewire, or no wiring diagram had been constructed and now there is a lot of guesswork involved in figuring out what cabling is where. Today, there is frequent addition or reconfiguration of personal computers or local area networks, which adds to the need to create additional flexibility.

Expense of Cabling: The expense of cabling can be as great as the cost of computer hardware and software. If the installer is fortunate enough to be working with a new building, drywall may not have been placed in the structure so that cabling is not only easier but less expensive. If the structure is complete, cabling can be quite expensive. Especially in older structures where stone forms the base of the construction, it can be a difficult environment in which to pull cable. These expenses are increased anytime that a move of a personal computer or a local area network takes place.

Physical or Regulatory Limitation on Cabling: Many types of cable will not guarantee signal strength if the cable length is over 1,000 feet. While repeaters (devices to "repeat" or push the signal further) can be used, there are still many limits on the physical distance which a signal can travel over coaxial, twisted-pair, or fiber optic cabling can travel.

Speed of Cabling Limitations: Cabling configurations can limit the speed at which the signal can travel. While token-ring configurations can support speeds of 4M and 16M bps and Ethernet networks can support 10M bps, sources in the wireless LAN market claim that it could easily achieve network speeds of 100M bps. Since the bandwidth of cabling determines the speeds at which signals can travel, physical limitations of bandwidth may constrain wired systems far below this level.

Need to Easily Form PC Workgroups: Local area network workgroups must often be put together very quickly, and often for short-term situations (such as a field audit performed by accountants). The cabling involved in quickly and economically constructing a workgroup, especially on a short-term basis, does not always make economic sense. While one could argue that cabling does not need to be concealed in the wall for short-term projects of this nature, others would argue that security and professional reasons mandate that the cabling be concealed. These scenarios are those for which wireless local area networks are ideal.

In fact, wireless LAN industry observers claim that the ideal local area network configuration can be best achieved through wireless techniques in the future. Network attributes commonly cited in this ideal wireless configuration are:

- easy to install and move
- can coexist with wire, cable, and future optical fiber
- easy to operate
- universally applicable
- can provide similar speeds to what it replaces
- secure
- reliable
- cost effective

Why Not to Use Wireless LANs

Conversely, there are a number of reasons to *not* use wireless LANs. They include the following:

Wireless LANs are Radio Devices. Are the electronic emissions from these devices necessary and safe?

Cost. With wireless technology in its early stages, some of the technology is still not cost effective.

Speed. With spread-spectrum, carrier current, and infrared technologies, speed can be slower than what can be achieved with comparable wired systems. The WIN technology is the main area where we see potential for great improvements in speed over wired systems.

Line-of-Sight Issues. Line-of-sight restrictions cause the technology to have physical limits.

Distance/Range Issues. Coupled with line-of-sight are the issues which surround the distance that a signal can travel. Especially in spread-spectrum technology, this distance may be adversely affected by weather.

Interference. Spread-spectrum technology, which is shared, is susceptible to interference from assorted household equipment including microwave ovens, garage door openers, and industrial meat cookers. Burglar alarms have even been found to emit and sense reflected radio waves blanketing out wireless signals.

Newness of Technology. As is the case with many new technologies, many users perceive wireless LAN technology to be untried. While this perception may not be valid, it is an issue which must be addressed.

Need for Special Software Drivers and Proprietary Components. Many wireless LANs require special software drivers and proprietary components in order to function. This requirement, by definition, locks the user into the vendor from which he or she purchased this proprietary technology.

AC Power-Line Product Restrictions. AC power-line products must be located on the same AC transformer due to the fact that low-frequency signals will not go through transformers.

Products and Services

Products

We have compiled a list of eight wireless LAN products currently available. This list, in Table 1, is by no means exhaustive but is intended to provide an overview of some products which are now being offered as wireless LANs.

Services

Advanced Radio Data Information Service (ARDIS)

ARDIS is a joint venture cellular-radio network between IBM and Motorola which is presently operating in 8,000 cities. It charges users for access to a slice of the radio spectrum which they are licensing from the FCC. An electronic mail and telecommunications program called MacKDT has been developed by Apple Computer to use ARDIS.

Data Personal Communications Services (PCS)

This proposed service is a bandwidth (1.7GHz to 2.3GHz) which Apple Computer has recommended that the FCC set aside for the free use of wireless LANs. The network would be accessible anywhere and would be free, yet it would carry only data.

Vendors

Listed here, for your convenience, are the addresses and telephone numbers of the vendors whose products are listed in Table 1.

BICC Technologies
1800 W. Park Drive
Westborough, MA 01581 (508) 898-2422

California Microwave, Inc.
985 Almanor Avenue
Sunnyvale, CA 94086 (408) 732-4000

Carrier Current Technologies Inc.
6505 Pharr Mill Road
Harrisburg, NC 29075 (800) 835-2402

Motorola, Inc.
3209 Wile Road
Arlington Heights, IL 60004 (800) 233-0877

NCR Corp.
1700 S. Patterson Boulevard
Dayton, OH 45479 (513) 445-5000

O'Neill Communications, Inc.
8601 Six Forks Road
Raleigh, NC 27615 (800) 624-5296

Photonics Corp.
200 E. Hacienda Avenue
Campbell, CA 95008 (408) 370-3033

Telesystems SLW, Inc.
85 Scarsdale Road, Suite 201
Don Mills, ON, Canada M3B 2R2 (416) 441-9966

Table 1. Wireless LAN Product

Product	Motorola Altair Wireless Ethernet	Telesystems SLW Arlan 440/450	Carrier Current Technologies CarrierNET Series	BICC InfraLAN
Technology	Wireless In-Building Network (WIN)/microwave	Spread-spectrum	AC power line	Infrared light
Applications	Wireless networking of PCs (with Ethernet cards), printers, terminals, other Ethernet devices	Wireless PC LANs; mixed combination of wireless and cable LANs; PC-to-host connections; network-to-network bridges	With Manylink software, file transfer, E-Mail, printer sharing via low-power, low-frequency radio signal; software does not support virtual drives	Traditional LAN applications
Protocols Supported	All running over IEEE 802.3 10M bps Ethernet	None identified	None identified	Ties into token-ring networks
Configuration	1 Control Module per Microcell; maximum of 32 User Modules per Microcell; maximum of 32 users per Microcell; maximum of 6 users per User Module	Arlan 440: wireless communications card with at least 1 Arlan 010 Transceiver over Arlan NCBus; up to 16 additional communications devices; Arlan 450: wireless communications card with attached or optional antenna; Requirements: 128K RAM per PC, Advanced NetWare 286 2.1 or ELS NetWare II	Requires 256K RAM and RS-232 serial port for each PC, DOS 2.0 or higher, 110/220 V AC, 50/60 Hz; 255 nodes on a power line up to 1 mile long	6-port repeater/multiaccess unit (MAU) for attaching PCs and bridges; 2 optical nodes transmit incoming and outgoing data signals
Frequency	18-19GHz	915MHz	200kHz	Optical wavelength: 870 nanometers
Range/Site Coverage	5,000 square feet (with 32 Ethernet devices); 40 feet between control and user modules	Indoor/office: 500 feet diameter; open factory: 3,000 feet diameter; building to building: 6 miles directional (line of sight)	Up to 1 mile	80 feet between nodes
Transmit Power	25 mW	1 watt, peak	5 watts	Not applicable
Signaling Rate	15M bps	230K bps	38,400 bps	4M/16M bps
Pricing	$3,995 (Control Module/32 users); $3,495 (User Module/daisy-chained 6 workstations, each with thin Ethernet adapter); net cost: $780 per port	$1,500 per node	CarrierNET: $199; CarrierNET Plus and CarrierNETPlusI: $399	$2,995 per node; under $500 per user
FCC License	Required	Not required but FCC certified under part 15.126	Not required	Not required
Strengths	Potential speed to 100M bps; security ensured due to data encoding; higher frequency leads to lower interference	Supports up to 100 PCs per network	Inexpensive	Minimal interference; inexpensive
Limitations	Proprietary technology not compatible with other network management packages; proprietary directional antenna required because signals fade rapidly; uses FCC-licensed frequency which costs the end user on an ongoing basis	Speed of network	Speed and distance constraints	Uses line-of-sight technology; cannot transmit through walls or partitions; relatively expensive at $2,995 per node

Table 1. Wireless LAN Product (Continued)

Product	O'Neill Communications Local Area Wireless Network (LAWN)	Photonics Photolink	California Microwave RadioLink	NCR WaveLAN
Technology	Spread-spectrum	Infrared light	Spread-spectrum	Spread-spectrum—differential quadrature phase shift keying (DQPSK)
Applications	File transfer, peripheral sharing, E-Mail	All traditional LAN applications	All traditional LAN applications	Runs on MS-DOS and Novell NetWare
Protocols Supported	AX.25 (X.25 standard for radio)	Compatible with TOPS (Sitka), PhoneNET, LocalTalk, or RS-232 links (4 computers)	RS-232-C (asynchronous/synchronous), RS-485, V.35, X.21 bis, IEEE 802.3, AppleTalk	Ethernet-type (CSMA/CD)
Configuration	LAWN transceiver unit; software requires 110K RAM (TSR program within software requires 40K); each PC requires 512K RAM and RS-232 serial port, DOS 2.0 or later; hard disk recommended	Infrared Photolink transmitter/receiver attaches to PC and desk or wall divider; 32 Photolinks at same target area; maximum of 128 computers on 1 Photolink network; Infrared Transceiver allows portable computers to be tied into Photolink network	RadioLink transceiver	Network interface card for ISA PC platforms, including network drivers and installation tools; external omnidirectional antenna module; optional DES encryption socket
Frequency	902-928MHz	Below visible light	902-928MHz; 2,400-2,483.5MHz	902-928MHz
Range/Site Coverage	1,000 square feet (coverage inside buildings); 500 feet (unobstructed in space); can extend another 500 feet using repeater (which works like microwave repeater tower that receives and resends telephone calls)	70-600 feet	Line-of-sight up to 5 miles; office building: 800 feet; dense building: 500 feet	Open environment: 800 feet; semi-closed environment: 250 feet; closed environment: 100 feet
Transmit Power	20 mW	25 mW	1 watt, peak	250 mW
Signaling Rate	38,400 bps	1M bps	250K bps	2M bps
Pricing	$499 per transceiver	4 Macintosh transmitter-/receivers: $1,195; Infrared Transceiver: $20 (OEM qty.)	Not available	Network interface card including Novell drivers and omnidirectional antenna: $1,390; DES (Data Encryption Standard) security feature: $90
FCC License	Not required	Not required but has Class A FCC certification	Not required (FCC Part 15.247)	Not required
Strengths	Inexpensive; 1-year warranty	Potential speed to 10M bps; secure; easy to install	Secure; easy to install	Fully compatible with Industry Standard Architecture (ISA); DES encryption socket allows for enhanced security; transmission not blocked by office partitions (except thick concrete or metal); can be connected to wired network via single PC on wired network
Limitations	Incompatible with NetWare and other LANS; can communicate with only one other network at same time	Supports only LocalTalk; relatively low speed; affected by weather; requires direct line of sight to operate	Speed constraints; interference possibilities	Relatively slow speed; signal may interfere with other spread-spectrum devices

■

Frame Relay Technology

In this report:

Datapro Summary

The demand for switched high-speed data services is now emerging as a major user requirement. Frame relay is a higher performance service compared to traditional X.25 packet switching and is positioned by the four key vendors in this market (Northern Telecom, Digital Equipment Corp., StrataCom, and cisco) as a LAN interconnection service. Frame relay technology is an evolutionary step beyond X.25 for improving packet network efficiency and accommodating more efficient applications, such as wide area interconnection of LANs at 56K bps and 1.544M bps rates. Networks based on frame relay provide communications at up to 2.048M bps (for Europe), bandwidth on demand, and multiple data sessions over a single access line.

Description

Traditional packet switching is a technology of the mid-1960s for solving the networking problems of that era. At that time, bandwidth was scarce and networks maximized the efficiency in transport. With the widespread availability of fiber cable, whose intrinsic traffic-carrying capacity has been doubling every two to three years, the need to maximize efficiency at the expense of end-to-end delay and switching node complexity is no longer imperative. The 1960s' bit error rate also left a lot to be desired, with 10^{-6} stretching the technical limit. Fiber now routinely provides 10^{-8} to 10^{-9}, and Forward Error Correction (FEC) can improve that further. Error-prone circuits necessitated complex error checking and recovery procedures at each network node.

Hence, X.25 packet standards assume that the transmission media is error prone. To guarantee an acceptable level of end-to-end quality, error management is performed at every link by a fairly sophisticated but resource-intensive link protocol of the High-level Data Link Control (HDLC) family. HDLC provides core functions, including frame delimiting, bit transparency, error checking (with Cyclic Redundancy Checking), and error recovery, and other functions.

In frame relay, error correction and flow control are handled at network end points. Frame relay accelerates the process of routing packets through a series of switches to a remote location by eliminating the need for each switch to check each packet it receives for errors before relaying it to the next switch. Instead, only the switch that receives a packet from a sending device and the one passing it to the receiving device check for errors.[1] Alternatively, that function can be relegated in its entirety to the end-users' customer premises equipment (CPE). This error treatment increases performance and reduces bandwidth requirements, which in turn can—in principle—reduce communications costs and decrease the number of packet handling devices needed in a network.[2]

Frame relay is a multiplexed data networking service supporting connectivity between CPE (such as routers and bridges)

—*By Daniel Minoli*

and, eventually (when the service is tariffed), between CPE and carrier networking equipment. Switch manufacturers are now in a selling mode to invest in the technology.[3,4] Frame relay will be implemented on such products as LAN bridges, routers, and T1 multiplexers. Frame relay is a connection-oriented technology supporting packets of variable length.

Frame relay can be considered as the successor to X.25 and, like X.25, it specifies the interface between customer computers and a network, whether public or private. This interface specification is described in principle in CCITT Recommendation I.122 (1988), although that specification is only a framework document. It must be further codified by a series of implementors' agreements, particularly for interoperability testing. Obviously, the early mistakes of noninterconnecting "X.25-conformant" networks must be avoided for the mainstream data comm community to accept the technology.

The cost of dedicated T1 facilities is still fairly high. A typical interexchange carrier (IXC) tariff is $1,800 plus $10 per mile for metropolitan range distances (0 to 50 miles), and $2,025 plus $7.50 per mile for regional or national distances (101 miles or more). In addition, one must add the cost of the local loops at both ends of the circuit. The cost of a local loop varies depending on location: it consists of a fixed portion plus a mileage portion. The charge typically ranges from $300 to $800, depending on carrier and mileage.

Therefore, using a frame relay-configured bridge or router over a dedicated T1 link is not advantageous. On the other hand, such an interface used in conjunction with a fast-packet backbone multiplexer could be cost effective, since the user can obtain from the backbone needed bandwidth on a demand, rather than on a preallocated (and inefficient) basis. StrataCom (Campbell, CA) has the market lead in frame relay.

Frame Relay versus Available Technologies

Initially, frame relay was developed by the CCITT as an ISDN packet mode bearer service with logically separate control plane (C-plane) and user plane (U-plane) information. In the C-plane, all signaling capabilities for call control, parameter negotiation, etc., were contemplated to be based on a set of protocols common to all ISDN telecommunication services. In the U-plane, the basic bearer service provided in I.122 is the unacknowledged order-preserving transfer of data units from the network side of one user-network interface to the network side of the other user-network interface. The frame relay frame format is based on CCITT Q.921 (which corresponds to ANSI T1.602). New enhancements of I.122 are under way, as discussed later.

According to U.S. spec T1.606,[5] the following frame relay applications are conceivable:

1. Block interactive data applications. An example is high-resolution graphics (e.g., high-resolution videotex and CAD/CAM). The main characteristics of this type of application are low delays and high throughput.
2. File transfer, intended for very large file transfers. Transit delay is not as critical for this application as it is for the first application. High throughput might be necessary to produce reasonable transfer times for large files.
3. Multiplexed low bit rate. The multiplexed low bit rate application exploits the multiplexing capabilities of the Layer 2 protocol to provide an economical access arrangement for a large number of low bit rate applications. The low bit rate sources may be multiplexed onto a channel by a Network Termination (ISDN) function.
4. Character-interactive traffic. An example of character- interactive traffic is text editing. The main characteristics of this type of application are short frames, low delays, and low throughput.

Of these four applications, the first two are feasible because of frame relay's channel speed. The last two, although also feasible in conjunction with the speed, are somewhat more poised to exploit frame relay's intrinsic features. Figure 1 depicts the positioning of frame relay in the larger context of available internetworking technologies. As shown, frame relay supports bursty traffic at medium speeds. Technologies competing with frame relay include T1 links, fractional T1, fractional T3, ISDN Primary Rate service, H0, H11, Switched DS1, and, finally, Switched Multimegabit Data Service (SMDS).

The frame relay interface is a mix of evolving ANSI and CCITT ISDN data link layer standards that could eventually eclipse the X.25 standards. Frame relay reduces the protocol processing overhead inherent in X.25 and is reasonably well suited for routing LAN traffic. Frame relay provides both a private and a switched virtual circuit. Permanent virtual circuits establish a fixed path through the network so the receiving end can quickly reassemble a file. X.25 virtual circuits can route packets over different circuits, meaning packets can be received out of order. This forces the receiving end to reorder the packets before it can reassemble the file.

To achieve frame relay's benefits, one should employ a fast-packet switch. Fast-packet switches employ statistical multiplexing techniques allowing a channel's entire bandwidth to be applied to the transmission, but they do not allocate bandwidths to users who do not require it. These switches can be located on customers' premises as T1 multiplexers that support a private fast-packet network. The switches can also be located at a carrier's central office. Although fast-packet switches can support nonframe relay traffic, together the two technologies can increase throughput between locations containing large amounts of bursty traffic.[5] Users with bursty traffic can find it advantageous to upgrade T1 equipment that uses time-division multiplexing with frame relay and fast-packet technologies. The drawback to TDM is that users must allocate the T1 circuit into individual channels, each supporting transmission of a specific data source. Since that bandwidth is allocated to only one user, it remains idle when the user does not need it.

A debate among interested parties now centers on frame relay versus cell relay, which is the underlying technology of Broadband ISDN. Some vendors have committed to frame relay; others have a program to advance cell relay; and others are pursuing both technologies. Some view frame relay as complementary, others as competitive.[4] Frame relay and cell relay are designed to meet different objectives and have therefore evolved in different directions. A simple categorization follows:[6]

Frame relay is a medium- to high-speed data interface for private networks which will be implemented in the next few years. Frame relay is being standardized at the DS1 rate.

Cell relay is a high- or very high-speed switching system that supports public networks; these systems will emerge in the mid-1990s. Cell relay is discussed in the context of Sonet, which transmits at bit rates from 51.840M bps to 2.4G bps (or even 13G bps, eventually). Metropolitan Area Network-based services, however, such as Switched Multimegabit Data Service (SMDS), will be available in the same general time frame as frame relay.

SMDS is a Bellcore-proposed, high-speed (DS1 to DS3), connectionless packet switched service providing LAN-like performance and features over a metropolitan area. SMDS provides for the exchange of variable-length data units up to a maximum of 9,188 octets. It is defined as a technology-independent service, whose early availability via MAN technology is envisioned for the 1991 to 1993 time frame. SMDS is expected to be one of the first switched broadband offerings and eventually will be supported by BISDN. SMDS's Subscriber-Network Interface (SNI) complies with IEEE 802.6.

Speed of Contemporary User Applications

Customer premises equipment such as bridges and routers have operated in the 1M to 3M bps range. The maximum Ethernet throughput can be 10M bps; the effective throughput, considering protocols, can be in the 1M to 3M bps range. And while bridges and routers are quoted as transmitting in the 10,000 to 12,000 packet-per-second neighborhood, these are usually packets with no data. Therefore, while frame relay can be adequate for some LAN internetworking applications, other applications—such as CAD/CAM, medical imaging, heavy-use desktop publishing, animation, etc.—could need the higher speeds provided by SMDS and cell relay technology. The next generation of bridges and routers will process 50,000 packets per second in 1991; by 1992, many companies will require 100,000 packets per second.[7]

IBM has introduced 16M bps token-ring systems, for which a DS1-rate bridge/router may not be adequate for some applications. FDDI routers and bridges must operate at much higher rates. These 100M bps systems could become more prevalent, since the FDDI standards are practically complete, and FDDI might be deliverable over twisted pair. In addition, FDDI now interworks with Sonet, implying that there may be an impetus to introducing FDDI routers/bridges. This, in turn, could require high-throughput internetworking. Users may wonder how well a 1M bps frame relay service can bridge LANs operating at 100M bps. Moreover, FDDI rates are too low for some users (for example, in supercomputer environments); routers capable of sustained speeds of 800M bps are already available. Cell relay must operate at a high speed to support public network traffic, which when aggregated can increase to several orders of magnitude greater than any individual customer or one or more bridge/router(s).

IBM users now remotely extend the mainframe channel, another major high-speed application (see Table 1). The IBM channel has traditionally operated at 3M bps per second; it was raised to 4.5M bps in the late 1980s, and again to 10M bps in 1990. It appears unlikely that frame relay could effectively support these needs; BISDN/ATM would be better suited.

Frame Relay Limits

Frame relay is considered part of narrowband ISDN. Many observers have already pinned their hopes for data on Broadband ISDN, after a 10-year search for data services applications over narrowband ISDN failed to identify major new opportunities.

Frame relay is often touted as an "ideal" way to link LANs over wide area backbone networks.[8] At a purely technical level, however, since frame relay is a connection-oriented technology and LANs are connectionless, a connectionless service is the ideal way to interconnect them. One also should avoid developing entire technologies and deploying networks catering to a single application, like LAN interconnection.

Frame relay improves traditional X.25 packet switching for data communications. Cell relay supports the sophisticated services likely present in a 1990s' organization, including data, voice, video, facsimile, high-quality image and graphics, and integrated messaging. Network managers will probably have to provide an integrated corporate communications infrastructure because of the two major business "discoveries" by senior executives in the 1980s:

1. Managing the network accounts for a large portion of the telecommunications expense. There is a strong desire to reduce network management complexity and financial burden. Integrated networks are easier to maintain and cheaper to manage. A smaller number of

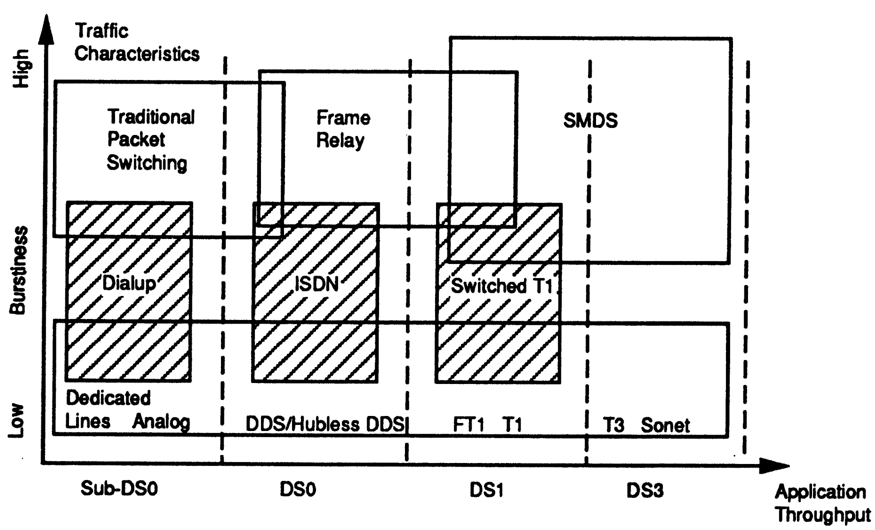


Figure 1.
Frame Relay in Context

Table 1. Applications of Channel Extender Technology

Application	Explanation
Distributed Data Processing	Different portions of the data can reside in different hosts' disk drives, spread throughout the country. With channel extenders(*), all hosts have quick access to all the information.
Instantaneous Data Backup	Large amounts of data can be dumped in realtime to a secure location mirroring all database operations. This eliminates the need to generate, manage, and ship tapes, which is a labor-intensive activity. This eliminates the current bottlenecks of occasional dumps of small selected data sets (or portions thereof) over slow communications lines.
Dual Center Transaction Processing	With appropriate programming, each transaction can be maintained in two or more hosts for fully mirrored operation with negligible delays.
Darkened Data Center	All the primary mainframe equipment is placed in a remote, secure, and darkened facility, possibly in a suburban area. Channel extenders can be used to connect remote users. Minimum personnel would be needed.

(*) Channels operate at 3-, 4.5-, and 10M bytes per second.

management systems should suffice, particularly with the commercialization of ISO-based network management standards.

2. Integrated networks cost less in terms of transmission facilities' recurring expenses because of the intrinsic economies of transport components. For example, for the cost of three to five DDS lines, one can already acquire a T1 facility. For the cost of three to five T1 lines, one can obtain a T3 facility.

Integration requires engineering simplification: Why establish parallel networks for each new service when they can all be supported on a common platform? Why go to all the trouble of designing, planning, installing, operating, and managing different networks like SNA, X.25, LANs, voice, etc.?

Consider a need to connect different user groups in two separate locations. For instance, a company located in Washington, DC, and Boston must connect similar islands of interest between the two locations: sales, engineering, and finance. Traffic between these three groups consists of four erlangs each at 0.01 blocking.

Employing the Erlang B model (for simplicity), the company would require 10 channels/trunks for each community, or 30 trunks between Washington and Boston. Assuming the two sites were 501 miles apart, the cost of 30 analog voice grade lines is $14,527 per month (for the IXC component); using two T1 lines would cost $11,564 per month. If the company can aggregate these three user communities, however, it needs only 20 channels for 12 erlangs of traffic at 0.01 blocking. This implies that it can use a single T1 line for only $5,782 per month, or 39% of the nonintegrated solution's cost.

Ultimately, such a choice hinges not merely on technologies, but on what users want, when they want it, and with what applications. As this report implies, frame relay institutionalizes another drawback of X.25 packet switching: it describes an interface specification. The equipment vendors can still utilize proprietary internal protocols, such as internal transport, routing, and flow control protocols, forcing a private network user to furnish the entire network with equipment from the same vendor. By contrast, the cell relay technology specified in the BISDN Asynchronous Transfer Mode is open by design.

Retrofitting a circuit switched T1 mux with frame relay interfaces does not deliver frame relay's intrinsic benefits. With circuit switching, the user must preallocate some (or, if desired, all) bandwidth to the frame relay service, whether the service uses that bandwidth or not. An efficient utilization of the technology over a private backbone network would require an internal fast-packet technology. By letting all applications compete for the backbone bandwidth, a frame relay application can access the entire bandwidth when it has data to transmit; a frame relay application on a circuit switched multiplexer can only access a fraction of the total bandwidth.[9]

Attempts to improve on X.25 have been under way since the early 1980s, with no apparent success. Several vendors pursued a concept called "burst switching." Users may be left wondering whether frame relay will follow the route of burst switching, which never materialzed.

A Protocol View of Frame Relay

Recommendations CCITT I.232 and I.462 (X.31) describe packet mode bearer services supported by an ISDN; I.462 (X.31) specifies the procedures for virtual call and permanent virtual circuit bearer services. ISDN also considers new packet-switching technologies in addition to these traditional X.25 packet modes. Three potential services proposed for standardization by the CCITT in Recommendation I.122 ("Framework for providing additional packet mode bearer services") are the following:

1. Frame relaying 1 (FR-1—no functions above core Data Link functions are terminated by the network; if needed, such functions are terminated only end to end);

2. Frame relaying 2 (FR-2—no functions above the core Data Link functions are terminated by the network; I.441 upper functions are terminated only at the end points); and
3. Frame switching (the full Recommendation I.441 protocol is terminated by the network).

FR-1 can be provided over permanent virtual circuits (PVCs) or switched virtual circuits. With PVCs, no call setup establishment is needed on a per-packet or session basis, since the address fields are agreed upon when the user subscribes to the service. In FR-1 the network has no knowledge of the end-to-end protocol. The LAP-D core functions include the following:

- Frame delimiting, alignment, and transparency.
- Frame multiplexing/demultiplexing using the address field.
- Inspection of the frame to ensure that it consists of an integer number of octets prior to zero bit insertion or following zero bit extraction.
- Inspection of the frame to ensure that it is neither too long nor too short.
- Detection of transmission errors.

Under ISDN, the new packet mode bearer services described above possess these characteristics:

- All C-plane procedures, if needed, are performed in a logically separate manner using protocol procedures integrated across all telecommunications services. Namely, Q.931 will be used to set up and tear down the service; the C-plane is used to establish global address mapping.
- The U-plane procedures share the same Layer 1 functions based on Recommendations I.430/I.431. Moreover, they share the same core procedures.

On the user side, Recommendation I.430 or I.431 provides Layer 1 protocol for the user (U-) control (C-) planes. The C-plane uses the D-channel with Recommendations I.441 and I.451 extended as Layer 2 and 3 protocols, respectively. In the case of permanent virtual circuits (PVCs), no realtime call establishment is necessary, and any parameters are negotiated at subscription time. The U-plane may use any channel on which the user implements at least the lower part (the core functions) of Recommendation I.441.

Frame Relaying Service

The term *relay* implies that the Layer 2 data frame is not terminated and/or processed at the end points of each network link, but is relayed to the destination, such as with a LAN. In contrast with CCITT's X.25-based packet switching, in Frame Relaying Service (FRS) the physical line between nodes contains multiple data links, identified by the address in the data link frame. FRS' major characteristics are out-of-band call control and link layer multiplexing. Unlike the (X.25-based) X.31 packet mode services, FRS integrates more completely with ISDN circuit mode services because of the connection control's out-of-band procedures.

It is expected that FRS, described in I.122, will become an ANSI standard, but additional supporting standardization is needed before the service can be offered in a carrier/vendor-independent fashion.

FRS is based on the ISDN D-channel LAP-D's frame structure, providing statistical multiplexing of different user datastreams within the Data Link Layer (Layer 2). In contrast, X.25 multiplexing occurs at the packet layer (Layer 3). In other words, FRS features the virtual circuit identifier, currently implemented in the Network Layer (Layer 3) of X.25, at the Link Layer (Layer 2) so that fast-packet switching can be accomplished more easily. In the X.25 environment, when a data call is established, the virtual circuit indicator is negotiated and used to route packets through the network for the duration of the call. The indicator is enveloped within the Layer 2 header/trailers, which must be processed before it can be exposed. This processing involves error detection and correction in addition to stripping the header/trailer. In the OSIRM environment layer, n+1 protocol information is enveloped inside layer information.

In LANs, actual packet routing is accomplished directly at Layer 2: the data packets are supplied with a 48-bit destination address, which is readily available and used to physically route the data to the intended destination. Also, no error recovery occurs in a LAN as a packet flows by a station on its way along the bus or ring. In FRS, only Layer 2's lower sublayer—with the core functions frame delimiting, multiplexing, and error detection—is terminated by a network at the user-network interface. Layer 2's upper procedural sublayer, which includes error recovery and flow control, operates between users on an end-to-end basis. In this sense, a user's data transfer protocol is transparent to a network.

Thus, FRS proposes to implement only the "core" functions of LAP-D on a link-by-link basis; the other functions, particularly error recovery, are performed on an end-to-end basis. Indeed, the capabilities provided by the Transport layer protocol (CCITT X.214/X.224, ISO 8072/8073) accommodate this transfer of responsibilities to the network boundaries. Frame relay service is a connection-oriented service, since routing is based on establishing virtual circuit indicators.

Figure 2 shows the partition of the Data Link Layer. For both FR-1 and FR-2, the network terminates only the core aspects of the data link protocol (I.441*). The terminals in FR-2 terminate the full data link protocol, whereas terminals in FR-1 terminate the "core" aspects of the data link protocol. What they terminate above core aspects is a user's option. See table associated with Figure 2.

Frame Relay uses routing based on the address carried in the frame; the frame itself is defined by CCITT I. 441. Figure 3 depicts the lower core LAP-D frame.[9] The "remainder" of the data link layer functions above the core functions must be defined into a peer-to-peer protocol. This protocol is indicated as I.441* in Figure 2.

Work is currently under way in the standards bodies (CCITT SG XI and ECSA T1S1). The addendum to T1.606 defines congestion management strategies; it covers both network and end-user mechanisms and responsibilities to avoid or recover from periods of congestion.[10]

Frame Relaying 1 Service Description

FR-1 data units are frames as defined in Recommendation I.441. The basic bearer service provided is the unacknowledged transfer of frames from S/T to S/T reference point. More specifically, in the U-plane it preserves their order as given at one S/T reference point if and when they are delivered at the other end (since the network does not terminate the upper part of I.441, sequence numbers are not kept by the network; networks should be implemented in

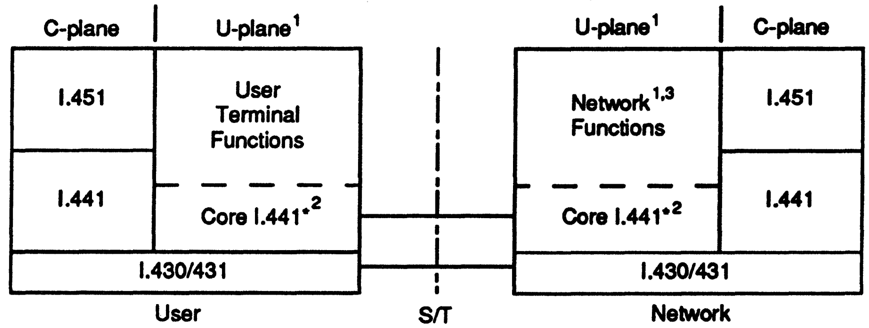


Figure 2.
User/Network Interface Protocol Architecture

1 The U-plane functions applicable to each bearer service are given in Table below.
2 The core functions of Recommendation I.441 are described in text.
3 The U-plane functions provided by the network at the S/T reference point are determined by the network after negotiation with the user, based on the requested bearer service and associated parameters. These functions are user selectable for each call. A network may choose not to implement the full set of options. These functions may not be available one by one. So far only three groupings have been identified:

a) the null set,
b) the upper part of Rec. I.441, and
c) the upper part of Rec. I.441 and the data transfer of X.25 PLP.

U-plane Functions Applicable to Each Bearer Service

Bearer Service	User Terminal (Note 1')	Network
Frame Relaying 1	I.441* Core (Note 2')	I.441* Core
Frame Relaying 2	I.441*	I.441* Core
Frame Switching	I.441*	I.441*
X.25-based Additional Packet Mode	I.441* X.25 DTP	I.441* X.25 DTP

Note 1' - Additional user-selectable functions may be implemented.
Note 2' - I.441* is I.441 with appropriate extensions. The use of the extensions may depend on each bearer service and is for further study according to the Blue Book.

such a way that, in principle, frame order is preserved); it detects transmission, format, and operational errors; frames are transported transparently (in the network), only the address and FCS fields may be modified (some bits being defined in the address field for congestion control can also be modified); and it does not acknowledge frames (within the network).

Frame Relaying 2 Service Description

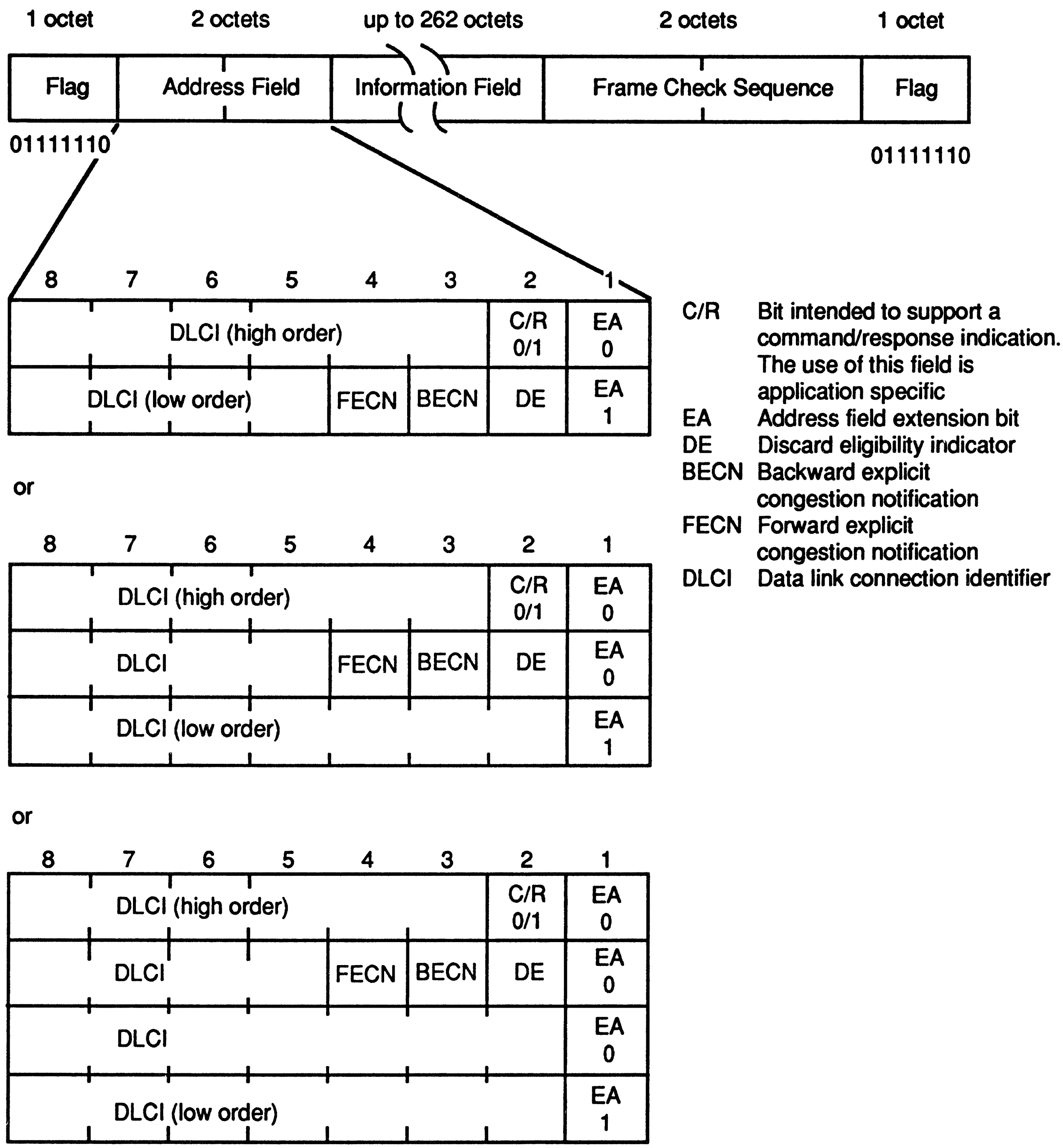
The frame relaying data units are defined in Recommendation I.441. The basic bearer service provided is an unacknowledged transfer of frames from S/T to S/T reference point. More specifically, in the U-plane it preserves their order as given at one S/T reference point if and when they are delivered at the other end (since the network does not terminate the upper part of I.441, sequence numbers are not kept by the network. Networks should be implemented in a way that, in principle, frame order is preserved); it detects transmission, format, and operational errors; frames are transported transparently in the network, only the address and FCS fields may be modified; it does not acknowledge frames (within the network); and normally, the only frames received by a user are those sent by the distant user.

Frame Relay Network Interworking

In the future, FRS may be provided by different carriers. There will then be a demand to interconnect these FR networks with existing Packet Switched Public Data Networks (PSPDNs). Any interconnection should allow users on different networks to communicate with each other in a uniform way, as though they were all on a single network.

To achieve these goals, standardized procedures must be established for the interworking between 1) FR networks (FR to FR) and 2) FR networks and X.25-based networks (FR to X.25). An FR connection may pass through multiple networks, including both private and public networks. For public networks, both local exchange and interexchange carriers may be involved. In a multivendor environment, the network of a given carrier or provider

Figure 3.
Address Field Format

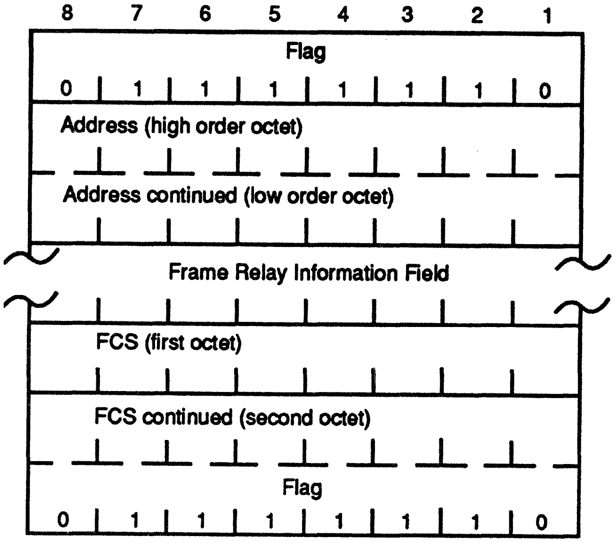


may consist of equipment from different vendors and/or different types of equipment from the the same vendor.

Two approaches for interworking can be used: network-to-network (NW-NW) interworking and node-to-node (ND-ND) interworking. In the NW-NW method, the signaling and transport procedures at an internetwork interface are standardized. Within a given network, internodal signaling is an internal matter.

Traditionally, telephone exchanges in a public switched telephone network are interconnected using the ND-ND method. This is the philosophy behind Signaling System No. 7 (SS7). In contrast, the NW-NW approach has been used for the interworking of data communication networks.

Figure 4.
Frame Relay Frame Format



Note: The default address field length is 2 octets. I t may be extended to either 3 or 4 octets by subscription.

ANSI Frame Relay Standardization Efforts

The American National Standards Institute (ANSI) has recently issued three draft documents in reference to frame relay in the U.S.:

1. T1.606: Frame Relaying Bearer Service—Architectural Framework and service description (T1S1/88-225)
2. Addendum to T1.606: Frame Relaying Bearer Service—Architectural Framework and service description (T1S1/90-175)
3. T1.6ca: Core Aspects of Frame Protocol for use with Frame Relay Bearer Service (T1S1/90-214)

The data transfer phase of the frame relay bearer service is defined in T1.606. This document specifies a framework for frame relaying service in user-network interface requirements and internetworking requirements. This document includes both interworking with X.25 and interworking between frame relaying service.

The protocol needed to support frame relay is defined in T1.6ca; the protocol operates at the lowest sublayer of the data link layer of the OSI reference model and is based on a subset of T1.602 (LAP-D) called "core aspects."

The frame relay data transfer protocol defined in T1.6ca is intended to support multiple, simultaneous, end-user protocols within a single physical channel. This protocol provides transparent transfer of user data and does not restrict the contents, format, or coding of the information or interpret the structure. This standard is applicable to Frame Relay Bearer Service (FRBS). It is intended for use on any bearer channels and operates on the D-channel concurrently with ANSI Standard T1.602.

Frame Relay Frame Structure

The frame relay frame format is shown in Figure 4. The field shown in the figure is described below.

Flag Sequence

All frames start and end with the flag sequence, consisting of one 0 bit followed by six contiguous 1 bits and one 0 bit. The flag preceding the address field is defined as the opening flag. The flag following the frame check sequence (FCS) field is defined as the closing flag. The closing flag can also serve as the opening and must be capable of receiving one or more consecutive flags.

Address Field

The address field consists of at least two octets, as illustrated in Figure 3, but may optionally be extended up to 4 octets.

Control Field

There is no control field for Frame Relay core services.

Frame Relay Information Field

The Frame Relay information field follows the address field and precedes the frame check sequence. The contents of the user data field consists of an integral number of octets (no partial octets). Networks can support a default maximum information field size of 262 octets. All other maximum values are negotiated between users and networks and between networks. T1S1 strongly recommends that networks support a negotiated maximum value of at least 1,600 octets for applications such as LAN interconnect, to prevent the need for segmentation and reassembly of user equipment.

Transparency

A transmitting data link layer entity must examine the frame content between the opening and closing flag sequences (address, Frame Relay information, and FCS fields), and must insert a 0 bit after all sequences of five contiguous 1 bits (including the last five bits of the FCS) to ensure that a flag or an abort sequence is not simulated within the frame. A receiving data link layer entity must examine the frame contents between the opening and closing flag, which directly follows five contiguous 1 bits.

Frame Checking Sequence (FCS) Field

The FCS field is a 16-bit sequence.

Order of Bit Transmission

The octets are transmitted in ascending numerical order; inside an octet bit 1 is the first bit to be transmitted.

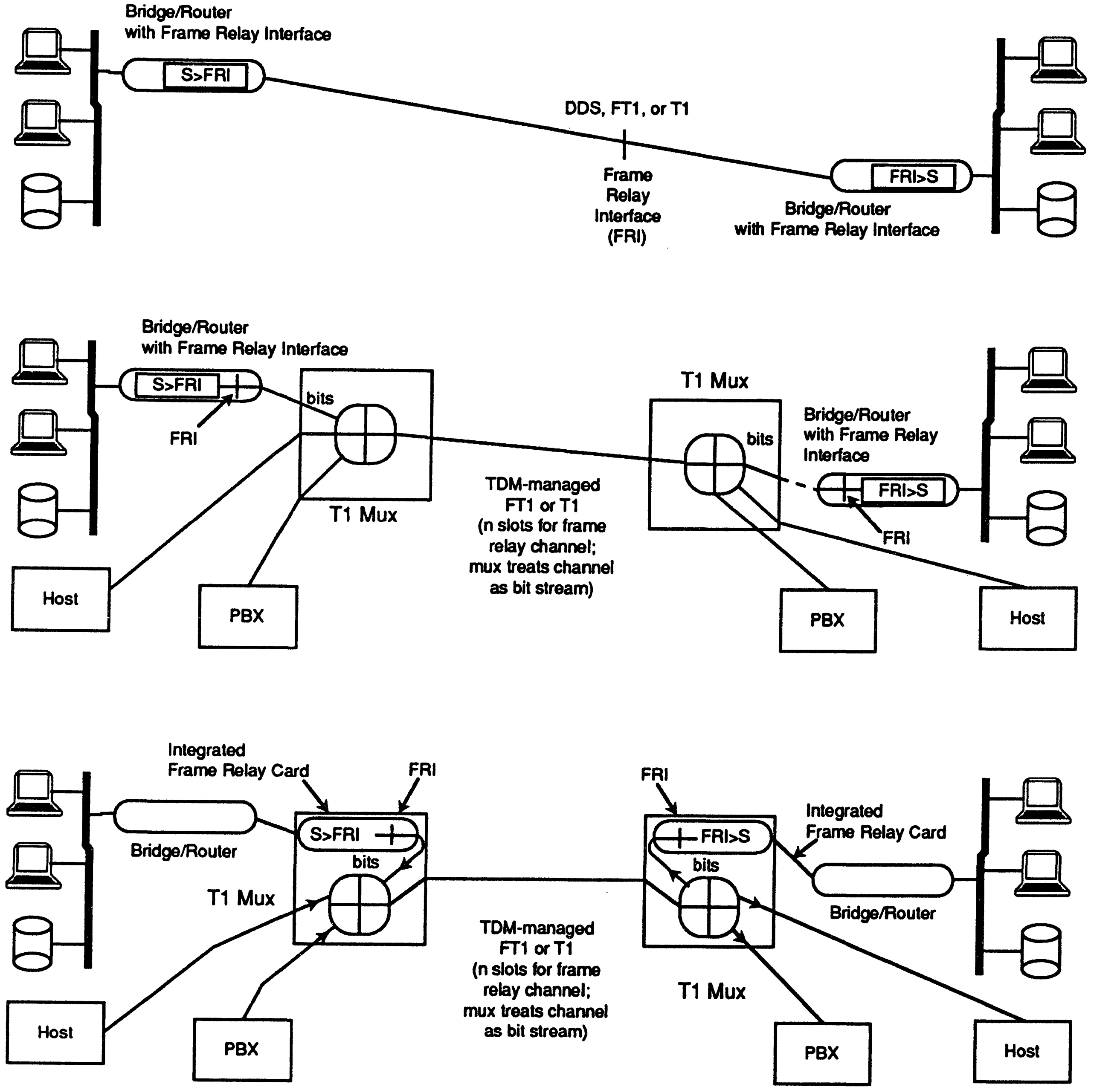
Invalid Frames

An invalid frame is one of the following:

- Is not properly bounded by two flags (e.g., a frame abort), or
- Has fewer than five octets between flags (Note: if there is no information field, the frame has four octets and the frame will be considered invalid), or
- Does not consist of an integral number of octets prior to zero bit insertion or following zero bit extraction, or
- Contains a frame check sequence error, or
- Contains a single octet address field, or
- Contains a Data Link Connection Identifier (DLCI) that is not supported by the receiver.

If the frame received by the network is too long, the network can do one of the following:

Figure 5.
LAN Interconnection Options



- Discard the frame, or
- Send part of the frame toward the destination user, then abort the frame, or
- Send the frame toward the destination user with valid FCS.

Selecting one or more of these behaviors is an option for designers of frame relay network equipment, and is not subject to further standardization by T1S1. Users cannot make any assumption as to which of these actions the network will take. In addition, the network may optionally clear the frame relay call if the number or frequency of too-long frames exceeds a network-specified threshold. Invalid frames are discarded without notifying the sender. No action is taken as a result of those frames.

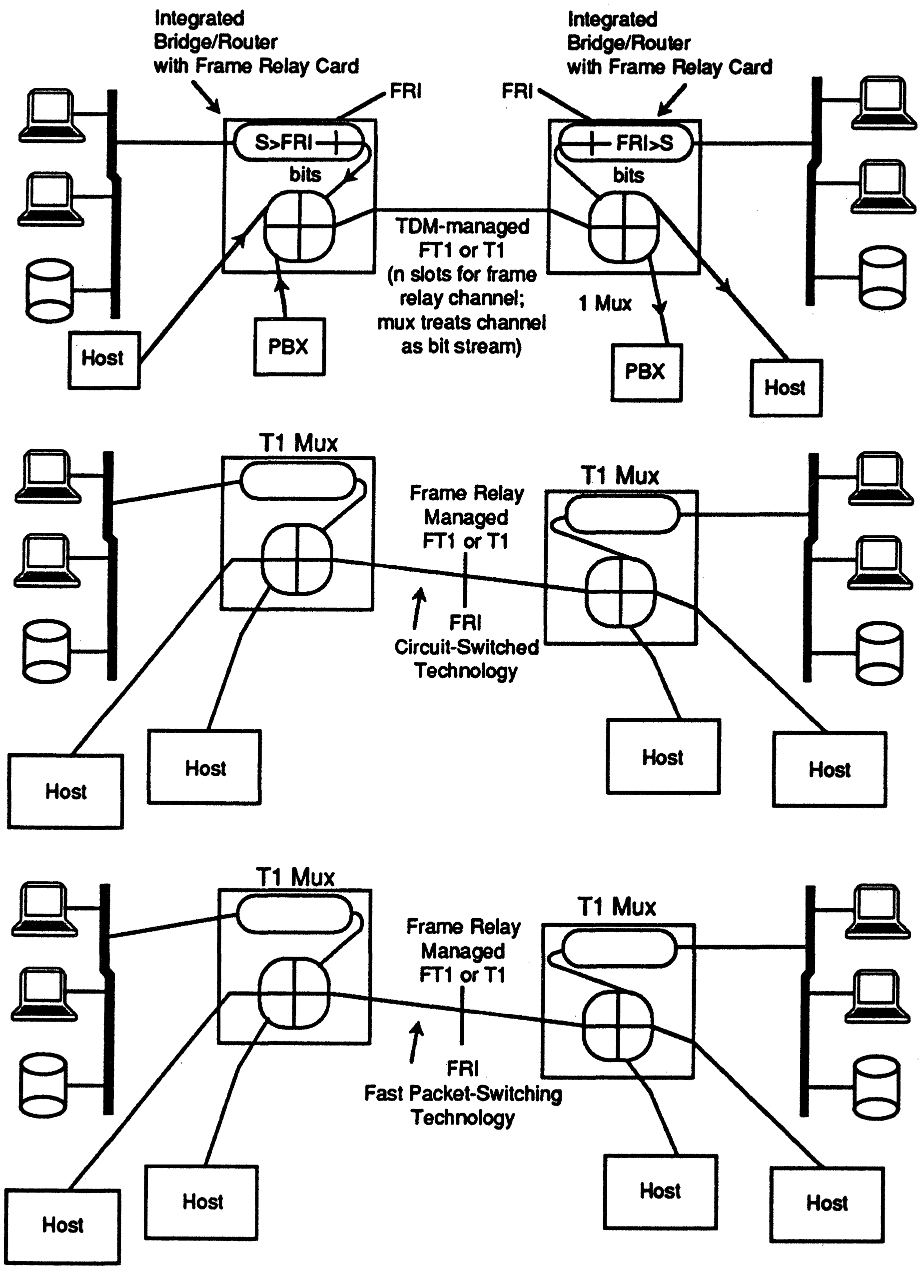
Frame Abort

Receipt of seven or more contiguous 1 bits is interpreted as an abort and the data link layer ignores the frame currently being received.

Address Field Format

The format of the default address field is shown in Figure 3. This field includes the address field extension bits, a bit

Figure 6.
Additional LAN Interconnection Options



reserved for use by end-user equipment intended to support a Command/Response indication bit, Forward and Backward explicit congestion indicator bits, a Discard eligibility indicator, and a Data Link Identification (DLCI) field. The minimum and default length of the address field is two octets, and it can be extended to three or four octets. To support a larger DLCI address range, the three-octet or four-octet address fields can be supported at the user-network interface or the network-network interface based on bilateral agreement.

Interim Specification

The ANSI standards were expected to be approved in mid-1991, and the CCITT standard would be finalized six months later.[11] As a consequence, at the end of 1990 a number of vendors backed an interim joint frame relay specification to ensure some degree of interoperability of new products delivered before ECSA (T1S1) and CCITT standards are finalized. The joint specification is not as complete as the ECSA draft document, but provides a basic set of agreements to begin developing products.

The need to offer interoperable frame relay products is critical, and vendors realize that users may not be willing to deploy technologies that lock them in with systems that could become obsolete in a year or two. cisco Systems, Inc.; Digital Equipment; Northern Telecom, Inc.; and StrataCom, Inc. jointly developed a frame relay specification on which product development can be based until international standards become available.[12,13] The interim specification, announced on September 4, 1990, is based

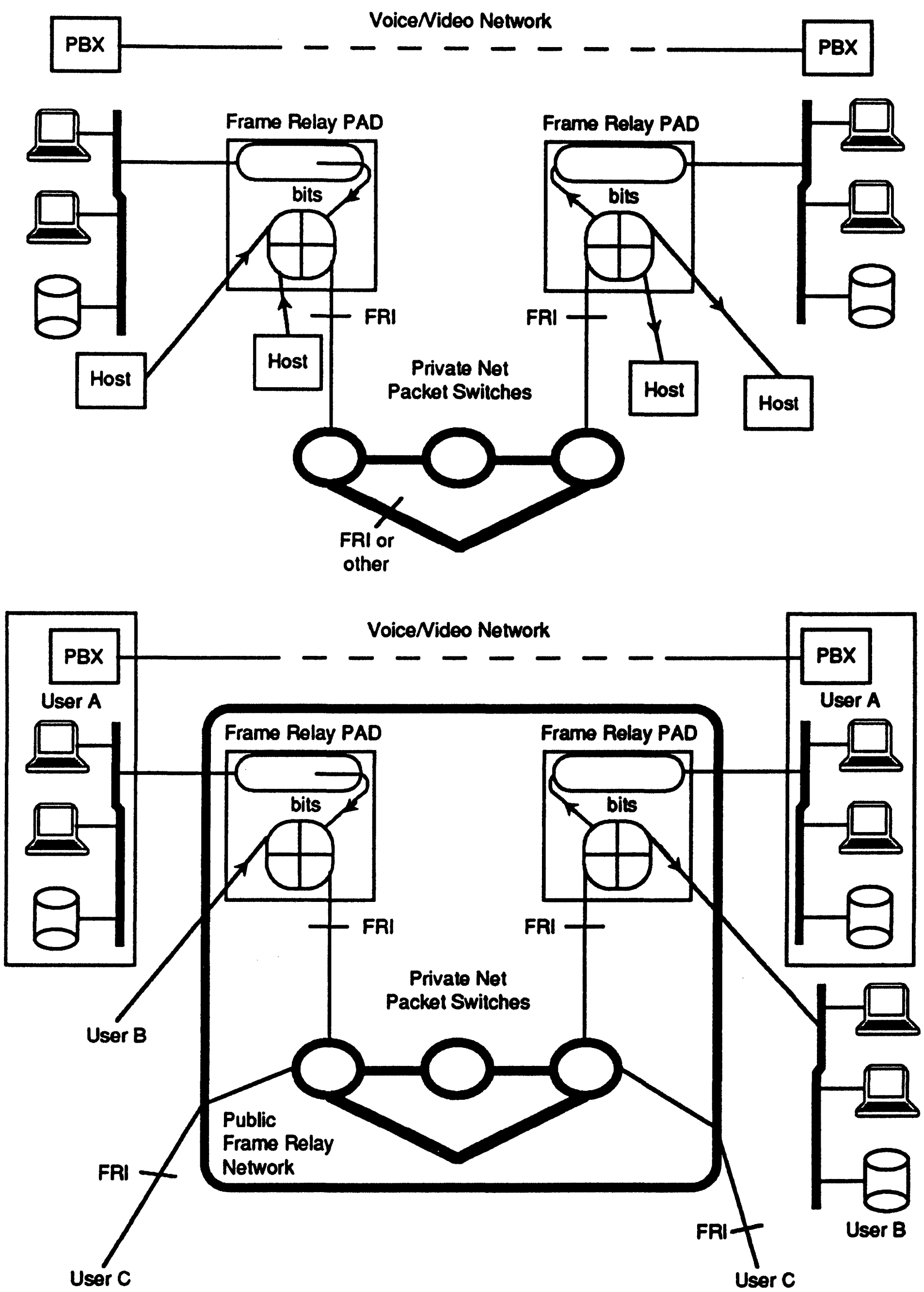


Figure 7.
Frame Relay Interconnection Options

on the emerging ECSA standard but has additional management features and broadcasting.[13] For example, it includes a 16-bit header containing data for routing and congestion control. It will also support automatic reconfiguration of devices with a frame relay interface, and has the capability to detect faults in devices using frame relay interfaces. The interim spec also covers the physical layer.[11]

Extensions to the basic T1S1 draft documents include the following:[14]

1. Support for a global addressing convention, to identify a specific end device
2. Multicast capability, to send frames to all devices which belong to a "multicast group"
3. Asynchronous status updates
4. Flow control, for preventing congestion collapse in a frame relay network
5. Extensions to the Local Management Interface (LMI)

StrataCom had been working on its own specification consistent with the ECSA and CCITT standards under development, but it agreed to share its data with the other three vendors when it decided to enter the joint effort.[11]

Approximately 20 other vendors have announced commitments to support the interim specification, including Newbridge Networks, Telematics International, 3Com Corp., Timeplex, Wellfleet, and Vitalink Communications Corp. Several vendors have started announcing product delivery dates; bridge and router vendors are among the first to announce frame relay support. By forming strategic alliances, vendors can get frame relay products to market more quickly.

Agreement on frame relay implementation specifications will facilitate the emergence of equipment from various vendors, allowing flexibility in user choices.[15] The equipment should also be usable with ISDN. Vendors are trying to avoid the implementation problems experienced in the early 1980s when X.25 packet-switching products entered the market—incompatible X.25 implementations still abound to this day. For example, with the interim specification, users of Digital's routers and cisco's line of multiprotocol router/bridges can connect their equipment to a StrataCom IPX-based private network, or to a public network with a) a Northern Telecom SuperNode at both COs serving the two end points, and b) a tariffed frame relay service.

For some vendors, such as those offering internetworking products, adding frame relay support may require a simple software upgrade since bridges and routers are already based on packet architectures; it will also require the addition of appropriately configured termination cards on the data comm side of the devices. Note, however, that MAC-layer bridges employ connectionless protocols; hence, the connection to frame relay is not trivial. Routers, on the other hand, may already support a connection-oriented service, and so the move to frame relay may be simpler. Routers are typically more expensive, however, than bridges.

Vendors of T1 multiplexers based on circuit-switching time-division multiplexing (TDM) architectures need more work to accommodate frame relay because they do not have experience with packet-switching technology. These vendors must add a packet engine to support frame relay.

Two T1 vendor approaches offer a short-term solution. The first approach is to develop frame relay modules, or boards, for existing circuit switched multiplexers. The second approach is to use a front-end frame relay developed by another vendor or strategic partner. With interim solutions, the T1 multiplexer may typically allocate only a finite amount of bandwidth for frame relay support, and performance and throughput problems may occur. In the long term, traditional T1 equipment may have to be redesigned to fully exploit the advantages of packet switching in general, and frame relay in particular.

Some vendors are reportedly concerned about the interim specification. The concerns center on possible anticompetitive implications. Some vendors may continue to pursue bilateral agreements on frame relay, raising the possibility of incompatible products, although everybody recognizes the detrimental overall effect of such a posture. Vendors including Advanced Computer Communications, Hughes Network Systems, and Netrix—all of which are actively working on frame relay products—reportedly would welcome the opportunity to help develop the interim specification. As of Fall 1990, another vendor group could possibly emerge with its own specification.

Other companies are calling on the interim specification architects to expand their group to bring in users and other vendors. Some vendors are also calling for a means of interoperability testing, an issue yet to be addressed. The standards bodies are pointing out that ANSI is the forum that users and vendors should use to work out a frame relay standard; the T1S1 version is already in draft form and in the process of final balloting.[15]

Frame Relay Applications in Private Networking

Users with dedicated LAN links today must examine traffic loads to determine if frame relay and fast-packet will be beneficial. Users with little LAN interconnection traffic may be better off using a 64K bps circuit on a TDM-based T1 multiplexer, while those with higher volumes may want to replace TDM multiplexers with ones supporting fast-packet and frame relay.

Because frame relay-based equipment better utilizes backbone facilities than existing circuit switching T1 multiplexers (and also existing X.25 switches), frame relay and fast-packet switching benefit users who want to connect LANs over integrated backbones supporting a variety of other traffic, assuming that these applications can also share the frame relay bandwidth (to take advantage of resource sharing). But users who simply want to provide high-speed links between remote LANs may be better off using fractional T1, T1, fractional T3, or even T3 links, until SMDS becomes available.

In the short term, most users who need LAN internetworking may find the preferred path is using bridges or routers linked via private lines, according to several analysts.[5] This approach provides excellent response time; the drawback is the cost of the dedicated facilities. Unless frame relay communication service is tariffed, and at the right level, using a frame relay-configured bridge over a dedicated link does not appear to add value, and, in fact, affects response time and involves expenditures for new equipment.

According to some observers, most users must transport a mix of data, voice, and video; hence, they will find it difficult to cost justify building a network solely dedicated to LAN traffic.[2]

Figure 5 shows some examples of LAN interconnection options using frame relay technology. The top portion shows the use of a T1 line dedicated to a new bridge/router system incorporating frame relay. The middle portion demonstrates where a fixed portion of bandwidth from a T1 mux is employed to connect a bridge/router system which incorporates frame relay. The bottom diagram shows a T1 mux with an integrated frame relay card but not a bridge; a fixed portion of bandwidth from a T1 mux is employed. These three scenarios are likely to represent the early usage of FR technology.

Figure 6 shows other examples of possible interconnection options using frame relay. The top part shows a T1 mux with an integrated bridge that uses frame relay; a fixed portion of bandwidth from a T1 mux is employed. The middle part depicts the situation where various streams run into the frame relay-configured mux; the trunk side uses frame relay, but the trunk bandwidth is managed in circuit mode. The bottom part is the same as the previous case, but the trunk side uses frame relay and the trunk bandwidth is managed in fast-packet mode.

Figure 7 depicts a more long-term usage of frame relay. The top part demonstrates a private packet switched network utilizing frame relay network-wide to achieve efficiency; PADs may be required. A separate network for voice and video is required. The bottom part shows a public packet switched network that utilizes and offers frame relay to achieve efficiency; PADs may be required. Multiple users share the network. A separate network for voice and video is required.

References

[1]J. Brown, "Frame Relay, Fast Packet Not Suited for All LAN Users," *Network World*, August 13, 1990, p. 2.
[2]B. Wallace, "US Sprint Unleashes Frame Relay Service," *Network World*, October 1, 1990, p. 1.
[3]*Communications Week*, September 24, 1990, p. 67.
[4]T. Sweeney, "Nynex is Leading the Way," *Communications Week.*

This report was prepared exclusively for Datapro by Daniel Minoli, adjunct assistant professor at New York University's Information Technology Institute and full-time data communications strategic planner. He recently published *Telecommunication Technology Handbook,* highlighting key communication technologies affecting the scene of the 1990s.

[5]T1.606: Frame Relaying Bearer Service—Architectural Framework and service description (T1S1/88-225).
[6]J. McQuillen, "Frame Relay vs. Cell Relay," *Business Communications Review*, 1990.
[7]J.M. McQuillan, "Broadband Networks—The end of Distance," *Data Communications*, June 1990.
[8]DSS1—Core Aspects of Frame Protocol for use with Frame Relay Bearer Service (T1S1/90-214).
[9]P. Desmond, "Frame Relay Standards Get T1 Mux Makers' Backing," *Network World*, June 11, 1990, p. 2.
[10]Addendum to T1.606: Frame Relaying Bearer Service—Architectural Framework and service description (T1S1/90-175).
[11]*Network World*, September 10, 1990, p. 1.
[12]*Network World*, October 15, 1990, p. 68.
[13]Joint News Release: Northern Telecom, Digital, StrataCom, cisco Systems, September 4, 1990.
[14]Frame relay specification with extensions, Revision 1.0, DEC, Northern Telecom, StrataCom, cisco, September 18, 1990.
[15]P. Desmond, "Firms question goals of frame relay group," *Network World*, October 1, 1990, p. 9. ■

A Killer Application for Frame Relay

In this report:

Datapro Summary

Frame relay is a fast packet technology that automatically reconfigures itself as it meets realtime network traffic conditions. Frame relay and other types of fast packet technologies derive their speed and power from traditional X.25 packet processing. They prevent network bottlenecks by adjusting their bandwidth as needed. Although frame relay is in its infancy, the time to begin planning for its network implementation is now.

Never before in telecommunications history has a technology been so quickly accepted as has frame relay technology. Two years ago, few people had even heard the term. Most of those who were familiar with the technology considered it to be in that category of "someday" technologies that we may never see. Now, out of nowhere, frame relay has become the hottest new buzzword. But is there any real meat behind the hype?

For a technology to be used widely, a "killer application" must exist. (A killer application is an overwhelming, compelling reason to adopt a new technology, either to save money on existing applications or to allow the implementation of applications which would not otherwise be possible.) T1 networks grew at an astounding rate in the second half of the 1980s because of the killer application of integrating voice and data transmission, resulting in vast savings in transmission bandwidth costs.

ISDN has failed to make tremendous inroads into the traditional telecommunications markets because it lacks a killer application. Few, if any, "mission-critical" tasks can be accomplished with ISDN that could not be addressed adequately without ISDN.

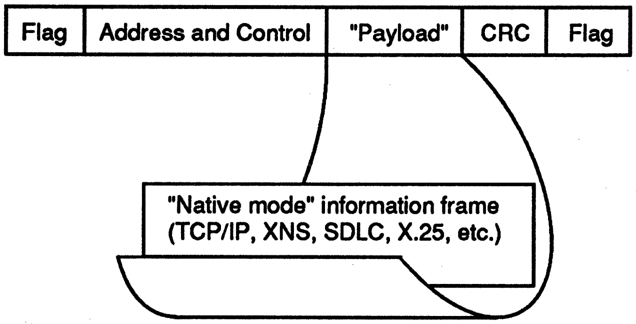
Frame relay already has a killer application in LAN internetworking. As users move along the path toward decentralized processing, local connectivity problems have been solved by attaching network elements—workstations, minicomputers, and microcomputers—via LANs. Now users want to have global communications that make any device anywhere appear as if it were attached to the local LAN. Providing this type of communications across wide area networks at several million bits per second presents the network manager and the network planner with a tremendous challenge.

Internetworking With a New Twist

Of course, there's nothing really new about LAN internetworking. For several years, LANs have been internetworked using various types of bridges, routers, and gateways. In fact the technology here is much better defined than the terminology. Although confusion continues to abound concerning when a bridge is a router, when a router is a bridge, and exactly which products do what, it is clear that many of these products work quite well. Moving traffic between two similar LANs is no longer a gargantuan

This Datapro report is a reprint of "A Killer Application for Frame Relay" by Steven A. Taylor, pp. 28-35, from *Networking Management*, Volume 9, Number 8, July 1991. Copyright © 1991 by PennWell Publishing Company. Reprinted with permission.

Figure 1.
Frame Relay Encapsulation Technique



Frame relay frames simply "wrap" around information frames in their native mode, much like the wrapper around a candy bar.

task, and even converting between dissimilar LANs is quickly becoming a reasonable task.

Problems continue to exist, however, in sizing the connections between LANs. When LANs are transferring traffic at 4 to 16 Mbps, connecting them with 56 kbps is seldom enough to avoid massive bottlenecks. It's like connecting Manhattan and New Jersey with a one-lane bridge. The basic path is there, and it's enough to handle light traffic loads. Although light terminal-mode traffic can be handled, a large file transfer can cause rush-hour-like congestion. More bandwidth is clearly needed.

In the other extreme, connecting two LANs with a full T1 could be like replacing a New England covered bridge with a massive, multilevel bridge with 16 lanes of traffic in each direction. It provides massive throughput for peak usage, but the vast majority of the capacity is seldom utilized.

What is necessary, instead, is a method for providing bandwidth as needed. This is not at all unlike the operation of toll booths on an expressway. When traffic flow is light, only a few booths are open so few resources (toll collectors) are needed. At rush hour, more booths are open to handle the demand. In fact, in the case of "directional" rush hours, it is quite likely that the booths in the center are opened in opposite directions to accommodate changing traffic conditions.

Unfortunately, LAN interconnect data is more bursty and less predictable than the number of cars that use a typical freeway during rush hour. If data traffic behaved according to the auto traffic model, a time division multiplexer with time-of-day reconfiguration would provide the necessary response. What is needed instead is instantaneous reconfiguration based on real-time traffic conditions—a statistical multiplexer of some type.

This is exactly the function that frame relay and the other "fast packet" networking technologies provide. In fact, they are often spoken of as private lines with an elastic cross section so that at one instance they may be 56 kbps and at a later instance, when the traffic load demands more bandwidth, they may be T1 lines.

Fast packet technologies use statistical multiple, a technique that works on the assumption that various users need bandwidth in bursts, not on a constant basis. Thus, even though the absolute overhead bandwidth is much higher than with a time division multiplexer, this inefficiency is vastly overcome by sharing the total bandwidth pool on a dynamic basis. Busy channels can take full advantage of the spare bandwidth created by idle channels.

All of the fast packet technologies—frame relay, cell relay, and 806.2 SMDS—derive speed and power from streamlining traditional X.25 packet processing. They eliminate the tasks, such as error detection and retransmission, that were necessary for transporting the asynchronous data for which X.25 networks were designed. Leaving these tasks to higher level protocols allows tremendous throughput to be achieved at a reasonable price.

Still, a burning question for network managers and planners remains. When will frame relay be ready for production use in LAN internetworking? To try to answer this question, you need to examine several areas.

Standards

One of the most obvious advantages of using frame relay is the availability of a common interface standard for the transport of virtually any type of framed traffic. This could include LAN traffic, X.25, IBM's SNA/SDLC, and, if it is appropriately packaged, even packet voice. The key to accomplishing this is the standards themselves.

In reality, the question of how many standards bodies voted on a particular set of words is relatively meaningless. The real test is whether there exists some set of rules by which multiple vendors may develop products that work with each other.

A group consisting of most of the major CPE vendors, with a show of hitherto unequaled cooperation, has formed an organization known as the Frame Relay Forum. This group has banded together to aid, not replace, the standards process and to address issues of interoperability which were not appropriate for the standards bodies.

The net result is that standards are far enough along for users to have a high degree of comfort that they are not buying into a proprietary solution. Because frame relay products tend to be implemented in software, minor modifications and even upgrades should be relatively straightforward and routine.

Remember, though, that the standards only specify the interface. They allow multiple types of traffic from multiple LANs and other types of traffic to share a common transmission network, but they do not provide for total interoperability of equipment.

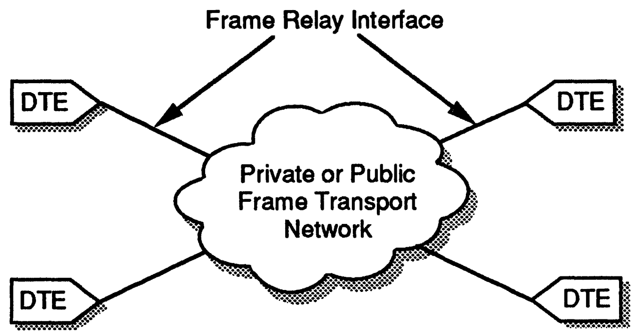
DTE Availability

Of course, a technology is useless if no products employ it. Once again, frame relay has a particularly good track record in this respect. Many manufacturers of equipment whose traffic could be transported over a frame relay network have announced support, and several products are already available.

The intrinsic simplicity of frame relay has contributed significantly to the rapid introduction of products. As Figure 1 shows, frame relay frames are built by encapsulating a frame of information in its native mode with a special frame relay frame. The information payload of the frame may be virtually anything, such as an Ethernet "packet," a TCP/IP "packet," an X.25 packet or frame, or even an SNA/SDLC frame.

Once this payload is formatted in its native mode, frame relay simply wraps the appropriate protocol around it, not unlike a candy wrapper around a candy bar. The variable-length information fields are a key component in frame relay's simplicity, since most information occurs in

Interfaces and Networks



One of the most confusing aspects of fast packet technology is determining exactly what the various technologies do, and what they don't do. This is particularly key in defining interfaces, DTE (to use the old, well-known data terminal equipment terminology), and networks.

As with X.25, frame relay specifications define an interface between a frame relay DTE (such as a bridge, router, gateway, host computer, or front-end processor) and a network. This interface specification is quite exact concerning data formats, congestion control, and other network management issues. However, it specifies nothing about how the data is to be transferred within the network. This is left as a network-specific implementation issue.

For example, whether the transport within the network is on a frame or cell basis is one of the most fundamental network-specific decisions. Each type of network will have some different performance characteristics, but neither is inherently better than the other in all respects.

The basic interface standard is the same whether the transport network is public or private. The only real difference is that a private network interface will generally be accomplished with a cable, while a public interface will typically use common carrier facilities such as 56-kbps or T1 circuits.

This interface standard allows a large number of diverse devices to share a common transmission fabric. However, it does not perform protocol conversion and other higher level tasks. These tasks must still be performed elsewhere within the network as necessary.

variable lengths in its native mode. The frame relay wrapper simply extends to the length of the payload. Only in rare cases will segmentation be required because the maximum negotiated length of the frame relay frame is less than the maximum frame length of the payload in its native mode. (By contrast, both BISDN ATM and 802.6 SMDS almost always require segmentation to fit variable-length native payload formats into their fixed-length cell format.)

The result of this simplicity is that frame relay can be implemented by the traditional CPE fairly easily. This has already been evidenced by the rapid support of frame relay capabilities. Furthermore, implementation with or without frame relay should be about the same price. Customers should immediately start asking for frame relay capabilities on LAN internetworking devices, X.25 packet switching gear, and other types of packetized terminal equipment, including equipment for SNA/SDLC networks.

Transport Network Availability

Of course, once data is framed in a frame relay compatible mode, a network must accept the frames and transport them to their ultimate destination. The transport mechanisms for these frames are not subject to the standards, so each network vendor's implementation will be different.

At this time, the transport network market has a few well-defined early entrants. This should change very quickly as more manufacturers have time to introduce products. The scope of manufacturers entering this market, which is very broad, includes traditional T1 multiplexer manufacturers, X.25 packet switch manufacturers, large PBX manufacturers, and start-up companies that are targeting this specific market.

Not only will equipment variety be broad, but users will be faced with choosing between a public network service and a private network implementation. In fact, some users may choose to use a public network initially so they won't have to choose among the available equipment for private networking until the market matures a bit.

All of the traditional private networking criteria for choosing equipment will continue to apply in this market. Network management features will be of utmost importance, since packet networks are intrinsically more complex than traditional T1 TDM networks. Key features will include congestion avoidance and management strategies, automatic alternate routing capabilities, and strategies to control network access. Standards compliance will be a given, but buyers must take care to determine which optional extensions to the various standards are supported.

Several carriers have already announced public frame relay services, and some services are already in production. To date, the majority of these services are intended for interexchange service. Both the traditional packet switching carriers and the IXCs are expected to participate heavily in this market.

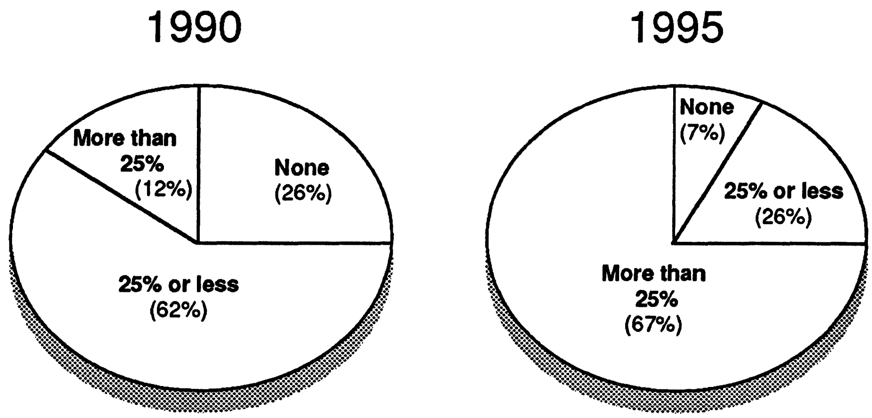
Because most of these services are not subject to formal tariffs, pricing is still a bit nebulous. However, pricing will ultimately determine the success or failure of these services.

At this point it appears that these frame relay services will be priced below their nearest technical competitor, fractional T1 services. It also appears that the trend is toward providing a fixed charge for a committed minimum level of service, rather than charging on a per-frame basis. This follows recent trends in pricing X.25 services. It is attractive to network managers and planners because they will know in advance what the service will cost.

Without this type of guarantee, network managers will be caught in the trap of using a "successful" service (people use it) that makes itself unsuccessful (costs go up with use).

Figure 2.
Percentage of Enterprises with WAN Traffic Originating on LANs

In a recent study detailing plans to move toward internetworking traffic in 1990, only 12% of networks studied had more than 25% of their traffic originating on LANs. By 1995, this number will climb to 67% with only 7% having no traffic originating on LANs.



Suffice it to say that pricing of these services will be a complex, albeit fascinating, task requiring extensive study.

The LECs have not heavily committed to providing frame relay services yet. On the one hand, you could be quite cynical and claim that this lack of commitment is primarily a result of fear of competition with SMDS, the LECs own preferred solution for LAN interconnection.

On the other hand, in the LECs' defense, you must analyze the need and applications for this technology in the intra-LATA versus inter-LATA context. Perhaps the economies offered by SMDS are more applicable as distance decreases. Nevertheless, it appears that frame relay services could be quite successful if they were offered.

In the final analysis, the same criteria that applied for years to X.25 networks will apply to frame relay networks. Private networks will tend to be more cost-effective between company locations with high traffic densities. Public services will tend to be more cost-effective where traffic density is lower and locations are more dispersed. Public services may also be more realistic for intercompany networks. Thus, the hybrid public/private network will prevail once again.

Other Technologies

The discussion so far has concentrated on the frame relay flavor of fast packet technology. This is entirely appropriate since frame relay technology will be widely available in the near term and should be quite competitively priced. Nevertheless, at least three other technologies will have an impact on the fast packet market.

SMDS networks based on the IEEE 802.6 standards are being promoted actively by several of the LECs as a method to provide metropolitan area network (MAN) services. This technology will serve LAN internetworking applications well, especially within the targeted market areas. Several CPE vendors are addressing equipment availability issues, and several trials of this technology are already in place.

The biggest unknown in the SMDS arena is pricing. Carriers can force the success of this service by pricing it attractively. They can also guarantee a lack of success by not pricing the services at an appropriate level.

"Real" broadband ISDN based on ATM (i.e., fast cell switching) will certainly play an important role in networks for years to come. Future frame relay services will be carried over high-speed backbones via ATM-based transport networks.

Certainly, the concept of having an ATM-based cell assembler/disassembler (the CAD being analogous to the X.25 PAD) on the premises as a part of the LAN internetworking device is very likely to be important at some point. This should provide very fast LANs, such as FDDI, with the significant internetworking advantages that frame relay currently offers to 4- and 16-Mbps LANs.

The question about ATM services is when they will become a reality, not if. SMDS services will depend on ATM for interexchange transport. The IXCs will probably offer this type of service as the logical extension of frame relay. Local access to these services may be via dedicated local access trunks (either provided by or bypassing the LEC), via SMDS services, or via ATM services that the LECs may decide to offer.

The third alternative technology that may play a key role is IBM's fast packet technology, which is known as "Paris." Although little information has been released about Paris, the few public domain documents seem to indicate that the technology is based on source-routed frame switching.

Source routing places a significant portion of the routing responsibility on the entry node, thus requiring the network only to execute the routing commands already placed in the frames. This allows higher speed network switching with less processing power in the network than with the other technologies examined thus far. Frame switching is more efficient than cell switching for most data transmissions since data tends to occur in random-length, not fixed-length, information entities.

Action Items

So, what is the prudent network manager/planner to do? At what point should you start moving toward these new technologies? While it might or might not be time to get out the checkbook, there are some immediate actions you should take.

- Educate yourself—The time has come to investigate frame relay actively and in a fairly detailed manner. Lots of products that could be keys to your strategic telecommunications planning either are already available or will soon be available. Frame relay internetworking can no longer be regarded as yet another idea that may become important. It's here, and it's very important.

As part of this process, you should also begin to learn more about SMDS and ATM, two other technologies that will ultimately be quite important. However, this technology is still a little ways off in most cases, so the education cycle here is not quite as critical.

- Define network needs and trends—Most processing environments are undergoing a gradual shift from host-centric to distributed computing (see Figure 2). This shift usually parallels a gradual migration from terminal-based systems to distributed workstations to locally networked systems to internetworked systems.

Obviously, this is not something that happens overnight. Each organization needs to establish a time line that plots where it is in this migratory path, and what it believes to be a reasonable progression over the next few years. Once this is established, the organization can develop a migration strategy to match its network to its information processing needs.

Perhaps you need fast packet technology now, perhaps in two years, perhaps later than that. Establishing this type of plan is not a commitment to implementing fast packet networking. However, the technology is sufficiently important that a conscious decision to plan or not to plan for it is becoming mandatory.

- Define the economic parameters—This decision, like almost all others, finally boils down to essential economic considerations. However, even the economic decisions have several levels.

On the basic level, the economic can be considered in the classic sense: "The current network costs are at a certain level, so I project that the new technology will save (or not save) a certain number of dollars per year."

Another level to consider could be called pro-active economics. Here, the current level of service is not necessarily a factor. Rather, it involves looking at what added capabilities could do for the network.

At this "what if" planning level, consider what additional functions could be added to the network if this new technology were available. Would these added capabilities provide a significant increase in productivity? What would be the best possible network if all available new technology were employed?

It is this level of planning that is both the most difficult and the most exciting. It often separates the networking leaders from the followers and gives the leaders a strategic advantage. As far as fast packet internetworking is concerned, the time to begin this planning is now. ■

Switched Multi-Megabit Data Service

In this report:

Datapro Summary

Switched Multi-Megabit Data Service (SMDS) is a proposed high-speed connectionless public packet switched service from Bellcore. SMDS, expected to be commercially available by the end of 1991, provides LAN internetworking on top of a universal addressing scheme and is based on the IEEE 802.6 standard. The report compares the features of SMDS with those of other packet-switching services technologies. The ongoing testing process for SMDS is reviewed.

Switched Multi-Megabit Data Service (SMDS) is a Bellcore proposal for high-speed (DS1-DS3), connectionless packet switched service that provides LAN-like performance and features over a metropolitan area. SMDS is a public packet switched service for exchanging variable-length data units up to a maximum of 9,188 octets.

Bellcore defines SMDS as a technology-independent service. It perceives the service as one of the first switched broadband offerings and projects eventual support of SMDS by BISDN.

Bellcore expects SMDS availability, via MAN technology, in the 1991 to 1993 time frame. Carriers will introduce SMDS as a public offering from the outset. As of late 1990, five BOCs—Bell Atlantic, BellSouth, Southwestern Bell, Pacific Bell, and NYNEX—had scheduled SMDS trials.

SMDS offers LAN internetworking on top of a universal addressing scheme.[1] The technology is being viewed as a prime candidate for building future networks, including gigabit-speed national research and education backbone networks.

At the protocol level, SMDS is based on the IEEE 802.6 standard, which now also interworks with the Broadband ISDN's asynchronous transfer mode (ATM) standard. ATM and SMDS use the same underlying switching technology, known as cell relay.

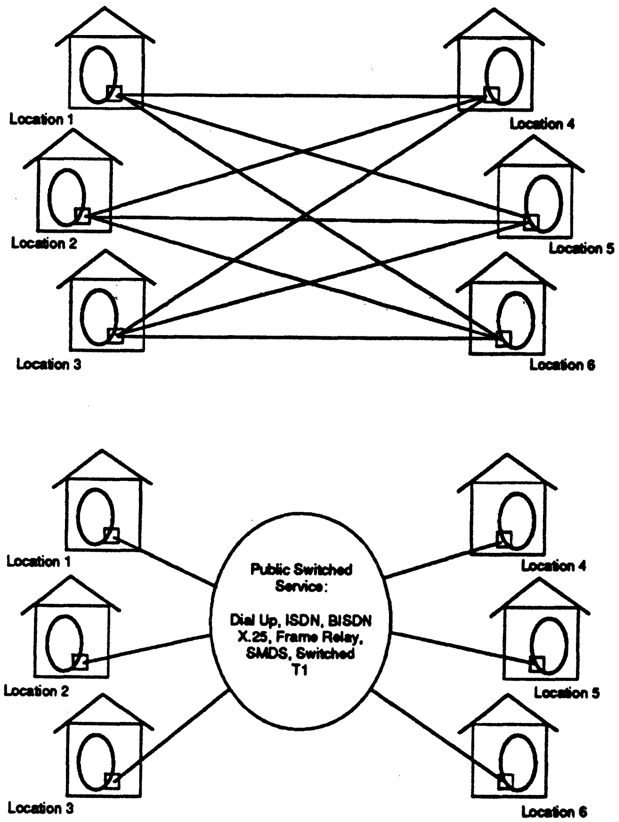
Cell relay segments data frames into fixed-length cells for transmission, with reassembly performed at the termination end; cell relay promises higher throughput than frame relay. Frame-relay technology is not based on either BISDN or IEEE 802.6, nor does it support speeds higher than 1.544M bps (2.048M bps in Europe).

SMDS is the first switched broadband service operating at 45M bps. Using cell-relay technology implies that SMDS will migrate smoothly to BISDN. The massive deployment of fiber in the public network will facilitate, if not expedite, SMDS deployment in the immediate future. Some carriers consider SMDS and frame relay complementary technologies.

The SMDS Connectivity Advantage

LAN interconnection technologies competing with SMDS include dedicated T1 links, Fractional T1, Fractional T3, T3, ISDN Primary Rate service, ISDN H0, ISDN H11, Switched DS1, traditional packet switching, frame relay, SMDS, and—in the mid-1990s—BISDN. Dedicated private-line networks become impractical when supporting a large number of remote data sources/sinks generating bursty traffic. As shown in Figure 1, the number of links grows quadratically with the number of interconnected sites. LAN installation growth is creating demand for public switched data services that deliver sufficient capacity, flexibility, and control (features that may

Figure 1.
Connectivity Advantage of Public Switched Networks



have in the past driven users to private network solutions), coupled with the universal access, survivability, economies of scale, and efficiency available through resource sharing. SMDS is ideally positioned technically to meet all of these requirements.

Users have employed dedicated lines operating at 19.2K bps, 56K bps, Fractional T1, and T1 speeds to interconnect LANs; speed increases are often dictated by applications requiring more information transactions, as well by the number of users utilizing the LANs. As the number of LANs grows, demand for high-capacity lines becomes expensive. A switched service is necessary to eliminate multiple physical paths between islands of users.

In addition to increasing LAN installations, however, the interconnection speed also increases. User equipment such as bridges and routers have operated in the 10M bps range for a number of years. Bridge and router vendors claim their units forward around 10,000 to 12,000 packets per second. The latest generation of bridges and routers processes 50,000 packets per second; by 1992, vendors might deliver equipment handling 100,000 packets per second. Thus, while frame relay may be adequate for some LAN internetworking applications, other applications will need the higher speeds provided by SMDS and cell-relay technology. Applications requiring cell relay include CAD/CAM applications, medical imaging, heavy-use desktop publishing, and animation.

The latest generation of LANs already requires higher speed interconnection. IBM has introduced 16M bps token-ring systems—a 1.544M bps bridge/router may be inadequate for these LANs. Fiber Distributed Data Interface (FDDI) routers and bridges must operate at much higher rates than existing systems. These 100M bps systems will become more prevalent now that FDDI standards are practically complete and vendors may actually deliver FDDI over twisted pair. Furthermore, FDDI now interworks with Sonet, creating additional user interest: users will not require dedicated multimode fiber to implement FDDI, but can use facilities from the public network. This may, in turn, require high throughput internetworking.

It is not certain that a 1.544M bps service can bridge LANs operating at 100M bps. For some applications, such as supercomputer networks, FDDI rates are too low. Cell relay must, therefore, operate at high speed because it is designed to support traffic on the public network. That traffic, when aggregated, might be several orders of magnitude greater than that of any individual customer or bridge/router.

Carriers are beginning to provide a number of public switched services and others are emerging. Carriers' frame-relay services support sufficiently high speeds. (Frame relay implemented in routers or even T1 muxes, however, does not by itself alleviate the problem of multiple T1 lines between dispersed sites.) Primary Rate ISDN is another viable solution. Switched T1 and Switched Fractional T1 provide a third solution (MCI has announced the availability of Switched T1 by late 1991). High-speed connectionless SMDS, however, may prove to be the best solution.

SMDS Market

SMDS is expected to be commercially available in the U.S. by late 1991 (by Bell Atlantic) and more widely available in 1992. A survey conducted by Bell Atlantic in 1990 of 275 data communications managers in manufacturing, health care, government, and other industries found that 40% to 70% were interested in introducing SMDS services in their (currently private) networks.[3] Trans-Formation, Tulsa, OK, recently released a forecast on SMDS lines. The company projects 29,000 lines installed by the year 2000.[4]

Bell Atlantic is trialing 45M bps SMDS access with Temple University; the test is reported to be proceeding well.[3] High-speed switching equipment from QPSX Communications is being used to interconnect 16 Ethernet LANs and two token-ring LANs at several campus sites over a diameter of 17 miles. Bell Atlantic service will initially be available in Baltimore; northern New Jersey; Philadelphia; Richmond; and Washington, DC.

BellSouth is reported to be planning internal SMDS use in 1991, using the service to link its LANs. New Hewlett-Packard workstations, servers, and networking products will be deployed in 140 BellSouth business centers, and will support about 13,000 employees.[5] All of the offices are planned to be connected by third-quarter 1991.

By year-end 1990, Pacific Bell had lined up five customers for an SMDS trial in northern California. The customers are Stanford University, Apple Computer, Tandem Computer, Sun Micro- systems, and Pacific Gas and Electric.[4] Pacific's six-month trial was scheduled to start in December 1990 and run through June 1991. In the current trial, SMDS will operate at the 1.544M bps access rate; at a later stage, the 45M bps rate may be trialed.

BellSouth, Pacific Bell, NYNEX, and Southwestern Bell announced plans to test SMDS based on available

AT&T Technology equipment. Bell Atlantic is moving ahead with a test of QPSX-provided MAN equipment. AT&T is reported to be aggressively positioning its equipment with the Bell Operating Companies (BOCs). The vendor has a prototype SMDS switch, called Superstar/BNS 2000, which uses the backplane of AT&T's earlier Datakit product (the backplane has a throughput of 200M bps).[6]

A major SMDS demo was held at Interop 1990, in October 1990. With Pacific Bell (as network manager), BellSouth, Southwestern Bell, NYNEX, Bell Atlantic, and AT&T provided the first multinode demonstration, connecting five sites around the country.[1] A host of vendors also demonstrated their interoperability to the SMDS specification, including BBN Software Products, cisco, Kentrox, Cornerstone Technology, Hewlett-Packard, Lotus Development Corp., Medical Imaging Systems, and Sun Microsystems.[7] cisco Systems Inc. demonstrated an SMDS router developed in conjunction with ADC Kentrox Industries Inc.

There is a goal of national connectivity in the U.S. by 1993. International SMDS connectivity may also become a reality in 1993 or 1994. The European Telecommunications Standards Institute has set up a working group to study MANs, and has adopted most 802.6 standard elements, including many SMDS features. Both Germany and Italy expect to undertake MAN trials this year.[1]

SMDS tariffs remain a large issue. In early 1991, Bell Atlantic was in the final stages of drafting tariffs for the SMDS to be available later in the year. Initial indications are that at least some carriers will offer SMDS access at DS1 and DS3 rates, at 10% to 30% less than the cost of the corresponding point-to-point links. SMDS will also entail a recurring usage charge beyond 50 gigabits of monthly usage included as part of the access fee.[3]

Public Metropolitan Area Networks: The Concept Behind SMDS

Local area network media access protocols cannot accommodate networks with large geographic radii without decreasing efficiency. Metropolitan area network protocols, therefore, support physically larger networks. MANs must transmit faster than LANs, since they typically interconnect groups of LANs. MANs' geographic reach generally requires them to be public in order to be cost effective.

Public MANs interconnect the locations of many organizations. An exception would be a large campus environment, containing dozens of buildings belonging to a university, medical center, or industrial park. MAN providers also plan to carry voice and video.

Two requirements determine the type of protocols used in MANs—the treatment of stream traffic (voice and video) and security. To meet voice transit delay restrictions, MANs can support a maximum delay of two milliseconds per packet; none of the previously existing IEEE 802 standards meet this constraint, so the IEEE developed the 802.6 standard.

A private network interconnects a single company's premises and is, therefore, fairly secure. To tap into the information stream, a potential intruder must identify the one cable among hundreds that carries the information in question. In a LAN environment, every user's data passes through every node in the network. Public MAN operators, on the other hand, separate the MAN into a transport network and an access network. The transport network carrying all network traffic remains under the carriers' control.

Table 1. Services Provided by the Network Layer

	Connection-oriented	Connectionless
Typical protocol	ISO 8208 (X.25 Packet Level Protocol)	ISO 8473 Connectionless Network Protocol (CLNP)
Packet treatment	Packet layer sets up logical channel	Packets sent independently
	Each packet has a logical channel identifier	Each packet has complete addressing information
	Same virtual circuit for duration of the call	Packets can take totally different routes
Services:		
Packet sequencing	Yes	No
Flow control	Yes	No
Acknowledgments	Yes	No
Protocol type	Complex	Simple

Bridging equipment is installed at a vault or pole outside the customer's premises, or at a Central Office. Users connect to the transport network over an access network attached to the transport segment via a bridge. Data addressed to other users is ignored by the bridge; the bridge transfers data onto the access network only when addressed to the assigned user. The bridge also performs a closed-user-group function for data originating within a subnetwork LAN connected to the MAN: It forwards data onto the transport network only if addressed to a MAN destination and ignores data destined to locations on the local network.[2]

In addition to transmission protocols, MAN systems require major network management support, including security management, accounting management, and fault management.

SMDS Technology

IEEE Project 802.6: Standards for SMDS

IEEE Project 802.6 coordinates MAN standard definition efforts. Widespread MAN introduction depends on standard-based equipment that allows simple interconnection. The use of media access bridging is a relatively new development; although it does not conform precisely to the Open System Interconnection Basic Reference Model, it offers performance advantages relative to network layer bridging. The access network need not itself be based on the IEEE 802.6 protocol; it can, in fact, be an extension of the user's own bus- or token-based LAN or LANs.

Table 2. Interconnection Technologies

	Public/ private X.25 net	TDM-based T1 private net	Fast packet-switching T1 private net	Frame relay over un-muxed private line	Frame relay over TDM-based T1 private net	Frame relay over fast packet-switching T1	Public net-based frame relay	SMDS	ATM
Switching unit	packet	bytes or group of bits	(T1) frame	frame	frame	frame	frame	cell	cell
Bandwidth conservation (bandwidth used only when information is being sent)	yes	no	yes	no	no	yes	yes	yes	not at the user level
PVCs, implying multiplexed links over single port	yes	no	no	yes	yes	yes	yes	yes	yes
Error treatment	link-by-link	user protocol	routing: network info: user	detect: network correct: edge-to-edge or user-to-user	detect: network correct: edge-to-edge or user-to-user	detect: network correct: edge-to-edge or user-to-user	detect: network correct: edge-to-edge or user-to-user	routing: network info: user-to-user	routing: network info: edge-to-edge or user-to-user
Bursty data	yes with PAD	yes	yes with PAD	yes with PAD	yes with PAD	yes with PAD	yes with PAD	yes	yes
Ethernet/ Token LAN interconnection	marginal	yes	yes	yes	yes	yes	yes	yes	yes

The P802.6 group started work on MAN issues in 1982. Early supporters included the satellite and CATV industries. The satellite providers wanted to provide inexpensive high-speed links between earth stations and local customers; CATV operators wanted to deliver data over their TV networks.

A number of protocols and media were analyzed between 1982 and 1987. In 1987, the committee selected a dual-bus architecture proposed by Telecom Australia as the basis of the MAN standard. The system, formally known as Distributed Queue Dual Bus (DQDB), was invented by students at the University of Western Australia. In May 1987, a company called QPSX was founded and the technology further developed.

The ninth draft of the IEEE 802.6 standard, published in August 1989, modified the length of the payload to 48 octets to interwork more directly with the ATM BISDN standards. The fifteenth draft of the standard was published on October 1, 1990 and was approved on December 15, 1990. There was an appeal on the standard which was considered by the IEEE and denied. Final publication of the standard was due in June 1991, based on the final approved version of Draft 15.

The 802.6 standard defines a high-speed shared medium access protocol used over a dual unidirectional bus subnetwork (together, the two buses provide bidirectional connectivity). The standard specifies the Physical Layer and the DQDB Layer required to support the following:

- An LLC sublayer—a connectionless MAC sublayer service supports an LLC sublayer in a manner consistent with other IEEE 802 LANs.
- Isochronous Service Users—provided by a connection-oriented service that may be used to transport isochronous data—e.g., PCM digitized voice. (Isochronous events or signals recur at known, periodic time intervals.)
- Connection-oriented Data Service Users—supported by a connection-oriented service that can transport bursty data; for example, signaling.

A typical 802.6 MAN consists of interconnected DQDB subnetworks. Subnetwork interconnection can be via bridges, routers, or gateways. DQDB is a distributed multiaccess network supporting integrated communications. Various connectionless, connection-oriented, and isochronous datastreams share the total capacity of the subnetwork flexibly and equitably.

The bus uses a kind of distributed reservation method. A bus controller maintains a counter for both transmission paths. The control counter is incremented for each slot request arriving from the direction of the destination, and is decremented for each vacant slot toward the destination. If the counter is zero when it arrives at a station, that station can transmit data using the next vacant slot. If upstream station requests are pending, the counter has a nonzero value.

The controller determines the value of the counter (which continues to count for future requests) and separately counts down as vacant slots come by. These unfilled slots satisfy the requests already pending when the station decided to transmit. When the value has reached zero, the

Table 2. Interconnection Technologies (Continued)

	Public/ private X.25 net	TDM-based T1 private net	Fast packet-switching T1 private net	Frame relay over un-muxed private line	Frame relay over TDM-based T1 private net	Frame relay over fast packet-switching T1	Public net-based frame relay	SMDS	ATM
FDDI LAN interconnection	no	yes	no	no	no	no	no	yes	yes
High-speed isochronous data	no	yes	yes	yes	yes	yes	yes	yes	yes: best
CAD/CAM internetworking	no	yes	no	no	no	no	no	yes	yes
Mainframe channel extension	no	yes	no	no	no	no	no	yes	yes
Voice	no	yes	yes with PAD	yes with PAD	yes with PAD	yes with PAD	unlikely	not initially	yes
Image	marginal	yes	yes	yes	yes	yes	yes	yes	yes
Video	no	yes	marginal	marginal	marginal	marginal	unlikely	no	yes
Throughput (bps)	19.2K	45M	1.5M	1.5M	1.5M	1.5M	1.5M	1.5-45M	150-600M
Network delay	high	lower	low	low	low	low	low	low	lowest
Availability	now	now	now	1991	1991	1991	late 1991	1992	1993-1995

next vacant slot is available for use. Greedy terminals are kept under control by limiting the terminals to only one outstanding reservation.

Currently, the DQDB system operates on fiber at 1.5 or 45M bps in each direction (if necessary, coaxial cable and microwave segments of adequate bandwidth can also be used). The dual bus is physically star-wired at a central location. The cable can be brought back to a CO for physical maintenance. The scheme differs from earlier ring systems in that the network, although cabled as a ring, is logically a bus. In practical terms, this means that a designated network controller node does not pass incoming data as do the other nodes; this node serves as the logical beginning and logical end of the two buses. The end node generates the slot framing for each fiber cable (similar to the master node in a ring system). In this environment, packets need not be specifically removed from the network at the end of their journey.

A major advantage of the dual bus topology is fault tolerance. If a node or line segment fails, continuity is maintained by logically designating the last active node before the failed portion as the end of the bus, and letting the previous bus end become a regular node. The nodes on either side of the failure then assume bus-end function. In addition to this mechanism, bridging relays allow data to pass through failed nodes. The bus maintains a distributed queue, providing optimal delay management.

SMDS Access

SMDS' Subscriber-Network Interface (SNI) complies with IEEE 802.6. SMDS users access the service via a dedicated SNI. Initially, Bellcore defined the SNI as an electrical DS3 interface; Synchronous Optical Network (Sonet) compatibility will be added in the future. SMDS access facilities will be fiber based.

The SMDS Interface Protocol (SIP) defines how the subscriber gains access to the SMDS network across the SNI. The SIP is a connectionless, three-layer protocol. SIP functions include addressing, framing, and physical transport.

SIP Layer 3 associates the appropriate SMDS addressing information with the SMDS data unit passed from the user. Framing of the Layer 3 data unit for serial transmission across the SNI is performed by the SIP Layer 2 protocol. An error detection mechanism is also provided at Layer 2. The SIP Layer 1 protocol provides Physical Layer functions; i.e., bit-level transmission across the physical facilities.

Since SIP is a connectionless protocol, and since it does not perform explicit flow control or error correction with ARQ methods, it provides an interface to higher layer protocols similar to that available in most LANs. As a result, using SMDS means that attached CPE must provide these upper layer functions.

The network examines every data unit transferred via SMDS and validates that each unit's source address is legitimately assigned to the originating SNI. For customers requiring higher traffic security and privacy, SMDS includes features that allow customers to form "logical private networks." Two such features, destination address screening and source address screening, allow customers to restrict communications across an SNI.

SMDS uses addresses similar to telephone numbers with extensions. SMDS also includes a group-addressing capability analogous to LANs' multicasting feature. When an SMDS user addresses a data unit to a group address, SMDS delivers copies of the data unit to each destination address identified by that group address. SMDS users can also designate multiple addresses (up to 16) to a single SNI.

SMDS provides different "access classes" at service initiation; each access class supports different traffic characteristics by limiting sustained information flow and information flow burstiness. The SMDS network enforces information flow limits. Customers need not provide resources, such as buffers, beyond those necessary to support the subscribed traffic limits, and the network provider can allocate its resources consistent with actual user requirements.

Connectionless Aspect of SMDS

Every layer of the OSI Reference Model, except the Physical Layer, supports two basic forms of operation—Connection-oriented mode and connectionless mode. Connection-oriented service requires a connection establishment phase, a data transfer phase, and a connection termination phase; a logical connection is set up between end systems prior to data exchange (see Figure 2). These phases define the necessary sequence of events for successful data transmission. Connection-oriented service capabilities include data sequencing, flow control, and transparent error handling.

In a connectionless service, such as SMDS, each Protocol Data Unit is independently routed to the destination; no connection-establishment activities are required, since each data unit is independent of the previous or subsequent one. Connectionless-mode service transfers data units without regard to establishing or maintaining connections, as shown in Figure 2. The (N)-UNIT-DATA service element performs data transfer and provides transfer of discrete data units. Each data unit must, therefore, contain at least the addressing information and the data itself.

In connectionless mode, transmission delivery is uncertain due to the possibility of errors. This appears contrary to the goal of network design—users want to ensure that messages reach their destination. In reality, connectionless-mode communication simply shifts responsibility for message integrity to a higher layer, which checks integrity only once, rather than requiring checks at each lower layer. Alternatively, each data unit might contain the error recovery mechanism. Table 1 compares a connection-oriented Network Layer service to a connectionless-mode service.

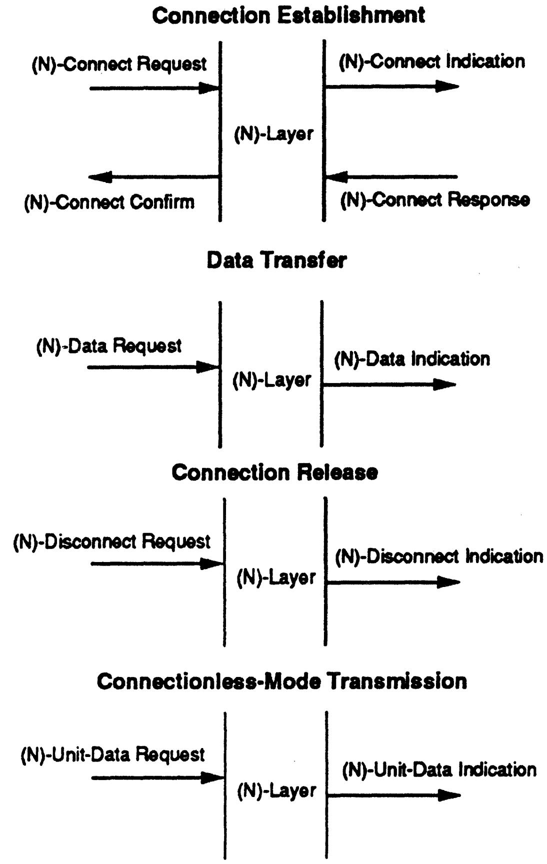
Each transmission mode represents the best approach for a specific niche. File transfers on the order of gigabytes might benefit from a connection-oriented-mode lower layer service, while point-of-sale (POS) inquiries might work best with a connectionless-mode service.

Generally, connectionless-mode data transmission may be ideal for:

1. Broadcast or multicast of information;
2. Inward data collection, which involves the periodic sampling of many sources (such as in process control); and
3. Transient processes (in military, aviation, and meteorological systems, there is often a frequent and abrupt dissociation from peer processes).

Connectionless communication at the lower layers of the OSI model is now well established and is found, for example, in local area networks (LANs) and metropolitan area networks (MANs). While the original OSI model—described in ISO 7498—was connection oriented, the ISO foresaw the need for connectionless service and issued an addendum to that protocol (ISO 7498/AD1). The ISO is now working to update the Connectionless Addendum and CCITT SG VII pursues a parallel process. The CCITT, however, has been reluctant to insert connectionless-mode data transmission concepts into CCITT X.200—its version of the OSI model. The ISO standard for Network Layer service, ISO 8348, contains connectionless service (in AD1) in addition to the connection mode.

Figure 2.
Connection-Oriented Mode versus Connectionless-Mode Communication

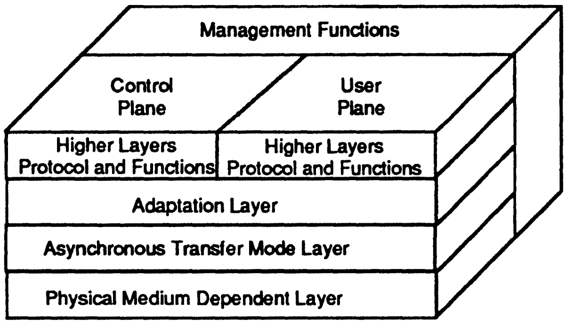


SMDS Relationship to Other Interconnection Technologies

Metropolitan area networks and SMDS use asynchronous transfer mode (ATM) technology, also known as "cell switching," at the Data Link Layer. ATM is an internationally agreed-to set of standards for high-bandwidth, low-delay, packet-based switching and multiplexing; it is a connection-oriented mechanism, though it is designed to support both connectionless and connection-oriented services. Transfer mode refers to the switching and multiplexing process; transfer mode logically resides on top of the Transmission Layer. ATM packetizes information to be transferred into fixed-size slots, called cells, which are 53 octets long. Cells are identified and switched by means of a label in the header.

An industry debate now centers on frame relay versus cell relay used by BISDN and SMDS. Some carriers and vendors have made commitments to frame relay; others plan to deploy cell relay; a handful of carriers are pursuing

Figure 3.
BISDN Protocol Model



both technologies. Some view the two approaches as complementary, others as competitive. Frame relay and cell relay are designed to meet different objectives and have, therefore, evolved in different directions. Some experts differentiate the approaches as follows:

- Frame relay is a medium- to high-speed data interface for private networks. Carriers will implement frame relay in the next couple of years. Frame relay is being standardized at the DS1 rate.
- Cell relay is a high- or very high-speed switching system capable of supporting public networks; these systems will emerge in the mid-1990s.

Metropolitan area network-based services such as SMDS, however, will become available at the same general time as frame relay.

Frame Relay

Current packet-switching technology, including fast packet switching and frame relay, will not be adequate for some applications. This is due to various services' characteristics, as well as the need to integrate them into a single access/transport technology. Applicable service characteristics include throughput, degree of burstiness, sensitivity to errors, and the capability (or lack thereof) to buffer the information in the network (i.e., sensitivity to delay). In comparison to establishing several dedicated networks, services and network integration has major advantages in terms of economic planning, development, implementation, operation, and maintenance. While dedicated networks require several distinct subscriber access lines, BISDN access can be based on a single optical fiber.

Fast packet-switching technology—as available with Stratacom IPX systems, for example—is an early vendor implementation of some of these integration principles, but is not the same as ATM. ATM cell size, cell structure, and header structure differ from those of fast packet switching. Vendors have demonstrated fast packet switching at 1.544M bps; cell relay is discussed in the context of Synchronous Optical Network (Sonet) transmission rate increments, starting at 51.840M bps, and reaching 2.4G bps (or even 13G bps, eventually). ATM is to operate at 150M and 600M bps. It is likely that to deliver Broadband ISDN services, new switching architectures, which will eclipse any fast packet-switching technology as we know it today, will emerge; photonic switching will likely play a major role in BISDN.

To meet the requirements of possible future broadband services, BISDN is designed to be flexible. The goal of BISDN is to provide a definition of the public network and user interfaces to the network that meets varied requirements, particularly when the traffic mix is highly dynamic. A variety of potential interactive and distribution broadband services is possible with BISDN:

- Broadband video telephony, videoconference, and video surveillance
- TV distribution (with existing TV and/or High Definition TV—HDTV)
- High-speed, unrestricted digital information transmission for high-speed LAN interconnection
- High-speed medical imaging
- Graphics-intensive Computer Aided Design/Computer Aided Manufacturing (CAD/CAM) applications
- Supercomputer networks
- Video document retrieval service (such as a videodisk library)

The BISDN protocol model is shown in Figure 3. The Adaptation Layer is service dependent. It supports higher layer functions of the user and control planes, and supports connections between BISDN and non-BISDN interfaces.

Information is mapped by the Adaptation Layer into lower layer cells. At the transmitting end, information units are segmented to be inserted into these cells. At the receiving end, information units are reassembled from the underlying cells. Any Adaptation Layer-specific information (e.g., data field length, time stamps, sequence number, etc.) that must be passed between peer adaptation layers is contained in the information field of the lower layer. Two BISDN access rates are being standardized: one at about 150M bps, and another at about 600M bps.

Asynchronous Transfer Mode (ATM)

The CCITT promotes ATM as the transport structure for future broadband telecommunication networks. For ATM, a number of functions of the Layer 2 protocol are removed to the edge of the network. Core Layer 2 capabilities are supported, in addition to Layer 1 functions (clocking, bit encoding, physical medium connection).

ATM is not an asynchronous transmission technique. The term "asynchronous" indicates that cells allocated to the same connection may exhibit an irregular recurrence pattern, as cells are filled according to the actual demand. ATM allows for bit rate allocation on demand, so the bit

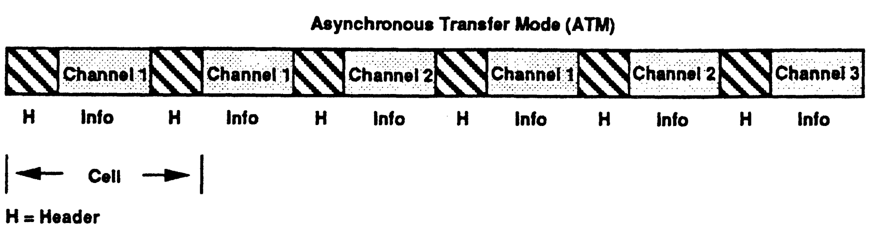
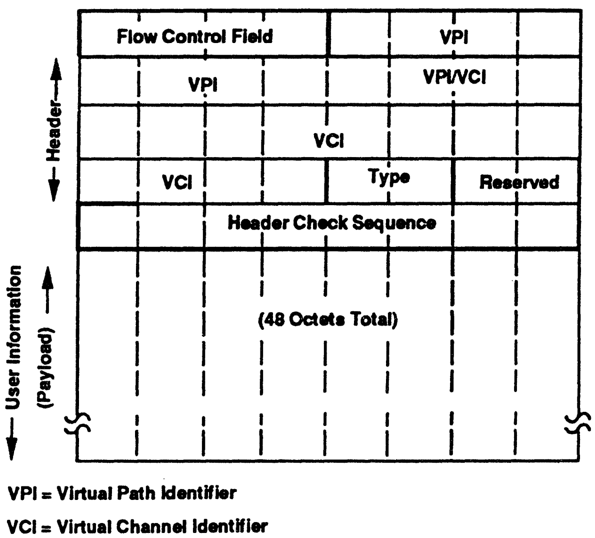


Figure 4.
STM and ATM Methods

Figure 5.
ATM Cell Format



rate per connection can be chosen flexibly. In addition, the actual "channel mix" at the broadband interface can dynamically change.

The ATM header contains the label, comprising a Virtual Path Indicator (VPI) and a Virtual Channel Identifier (VCI), and an error detection field, as shown in Figure 4. Error detection on the ATM level is confined to the header. The label is used for channel identification, in place of the positional methodology for assigning octets, inherent in the traditional synchronous digital systems. A point-to-point, connection-oriented service rides directly over the Transfer Layer mode.

ATM performs the relaying functions required to transport user information through the network. It should be noted, however, that ATM (or other transmission/switching) is somewhat hidden from the ultimate user, who only sees the interface. ATM technology can flexibly support a wide variety of services with different information transfer rates: it is suitable for both Constant Bit Rate and Variable Bit Rate Services (sometimes called isochronous and bursty streams, respectively).

ATM deals with procedures for allocating bandwidth at the user interface and to various user services. ATM is similar to packet switching, but with the following differences:[2]

1. No error control and flow control on the links—responsibility moved to the edge of the network.
2. Cells (packets) have a fixed, small length and no variable cells are allowed. This framework allows very high-speed switching nodes, since the logical decisions are straightforward. Cell length is now set at 53 bytes (48 for user information and 5 for overhead).
3. Unlike "packet switching" in LANs, ATM communication is connection oriented—all information is transferred via a virtual channel identifier that is assigned for the duration of the session.
4. The header provides only limited Layer 2 functionality. As a result, information transfer can be accomplished rapidly compared to other protocols; the protocols can be implemented in hardware.

The VPI/VCI is locally significant to the user interface, but may undergo translation as it travels to another interface. The VPI/VCI constitutes a label used to allocate transmission resources; the label, rather than the position of a frame, does the allocation. At call setup, the VCI value can be associated with a particular quality of service. Usable capacity can be dynamically assigned; the network can take advantage of statistical fluctuations while at the same time providing an established grade of service. Multiple rate-dependent overlay networks are obviated, facilitating integration.

The ATM cell's key features are as follows:

- ATM comprises a 5-octet header and 48-octet information field.
- The User Network Interface (UNI) header includes the following:
  - –A 4-bit generic flow control field, which may be used to achieve some of the functionality of an Access Control Field for multiaccess interfaces;
  - –A 24-bit label of which 8 to 12 bits can be used for the Virtual Path Identifier, and the remaining 12 to 16 bits are for the Virtual Channel Identifier;
  - –A 2-bit payload-type field;
  - –A 2-bit reserved field; and
  - –An 8-bit header error check field.

The VPI provides an explicit path identification at the interface; the VCI provides an explicit channel identification at the interface. At the Physical Layer BISDN will use Sonet. The UNI interface speed has been set at 155.520M bps. Thus, the information payload capacity in 125 microseconds is 2,346 octets, and the total information payload provided to the user is 135.979M bps.

Various Belgian carriers and manufacturers now plan ATM experiments. The experiment will last for five years, and the goal is to deploy a complete system, including terminals and transmission equipment, to assess ATM technology's viability. ATM multiplexers and switches are less dependent on the bandwidths involved in particular services. In theory, the label is the only ATM header field essential to perform switching and multiplexing functions.

Sonet

The relationship between ATM and Sonet, which is used by ATM, requires some explanation. Sonet is a Physical Layer (Layer 1) standard in its frame definition. Sonet is oblivious to what goes on beyond a single physical link, and so does not take an end-to-end routing view (although it allows straight-through connections to compose an end-to-end physical link). Sonet is a synchronous standard in the sense that the position of the information within the frame determines who owns that information. ATM is asynchronous in the sense that the position of the information does not establish ownership; a header field is added to each block of information to identify who owns the data in the block. ATM, as discussed, has a provision for virtual circuit indication, allowing end-to-end routing.

In packet switching, this routing function is a Layer 3 capability. In general, Data Link Layer routing is also allowed by Layer 2 protocols, although not commonly implemented in WANs (it is, however, common in LANs and is the basis for frame-relay service). ATM does aim at establishing an end-to-end path over which the bits can travel from origination to destination, which is intrinsic with the connection-oriented nature of virtual circuits and traditional telecommunications carrier philosophy. The routing capability of ATM can be interpreted as a Network Layer capability or a Link Layer capability; however, there is community interest in keeping ATM as physical and low level as possible, suggesting a Layer 2 interpretation of the ATM functions. This interpretation implies that ATM can use Sonet, at the lower layer, as a physical conduit.

There already are proposals in the Exchanges Carriers Standards Association (T1S1 in particular) to develop interworking interfaces for SMDS (IEEE 802.6), frame relay, and cell relay. The major technical hurdle is the interworking aspects associated with connection-oriented and connectionless association elements.

Comparing Interconnection Technologies

Frame relay fits in a continuum between private lines, SMDS, and BISDN. Frame relay describes an interface specification, and the equipment vendors can still utilize proprietary internal protocols. This is similar to CCITT X.25, where packet switches support a standardized interface but use proprietary transport, routing, and flow control protocols, which forces private network users to use a single vendor's equipment throughout the network. By contrast, the cell-relay technology specified in the Asynchronous Transfer Mode of BISDN is open by design.

If frame relay were available in the public network and it delivered user requirements for bandwidth, delay, network management, transition, reliability, and cost, it could be the vehicle to attract customers away from private networks and into the public arena, in preparation for BISDN. Hence frame relay can, in the view of these carriers, be viewed as a transition step to SMDS and BISDN (although the migration will require replacing equipment).

Some users are reportedly planning to incorporate frame-relay technology in their private networks, to determine if efficiencies can be achieved. Some experts predict that it is likely that frame-relay technology may evolve in the same way that X.25 did—appearing first in large, private networks and then moving slowly into the carriers' networks. A key consideration affecting the migration path (in addition to public availability and ubiquity) involves the cost and availability of user equipment to support a new service, such as frame relay. In the public arena, the progression of services in terms of complexity and availability will be frame relay, SMDS, and ATM/BISDN.

At the purely technical level, since frame relay is a connection-oriented technology and LANs are connectionless, connectionless network-based services (such as SMDS) provide the ideal way to interconnect them. Also, carriers want to avoid developing entire technologies and deploying networks that cater to a single application such as LAN interconnection. Frame relay, as currently being standardized, is designed for data communications only, as a long-overdue improvement of traditional X.25 packet switching. Cell relay is specifically designed to support the sophisticated mix of services likely to be present in a 1990s organization: data, voice, facsimile, high-quality image and graphics, integrated messaging, video, and HDTV.

Network managers will likely be asked by their organizations to provide an integrated corporate communications infrastructure. Integration affords engineering simplification: Why establish parallel networks for each new service when one can support all of them on a common platform? Why go to all the trouble designing, planning, installing, operating, and managing such different networks as SNA, X.25, LANs, voice, etc.?

Multiple service providers will have to be involved when frame-relay services or circuits cross LATA boundaries. Although standardization of frame-relay protocols makes the interworking between local exchange carriers, interexchange carriers, and international carriers feasible in principle, administrative, billing, and operational issues make the delivery of a nationwide service a massive effort, likely to take time. SMDS and frame relay fare better in this regard.

Table 2 compares X.25, TDM multiplexers, native frame relay, frame relay over a fast packet switch, SMDS, and ATM.

References

[1] T. Sweeney, "A National Handshake," *Communications Week*, October 8, 1990.
[2] D. Minoli, *Telecommunications Technology Handbook*, Artech House, Norwood, MA, 1991.
[3] E. Messmer, "Bell Atlantic Offers Glimpse at SMDS Pricing," *Network World*, November 12, 1990.
[4] S. Titch, "Four More Users Sign Up in Pac Bell SMDS Trial," *Telephony*, November 5, 1990.
[5] D. Bushaus, "Telco to Use SMDS," *Communications Week*, October 22, 1990.
[6] S. Titch, "SW Bell Plans SMDS Lab Trial," *Telephony*, September 24, 1990.
[7] T. Sweeney, "SMDS Showcase," *Communications Week*, October 15, 1990. ■

FDDI: Chapter Two

In this report:

Datapro Summary

Fiber distributed data interface (FDDI) offers the communications industry a great many benefits, including relief of bandwidth traffic, enhanced reliability, improved security, and more effective multimedia support. The market anxiously awaits FDDI's standards approval from the American National Standards Institute (ANSI). In the meantime, vendors and users are rallying to ensure that FDDI's market arrival is a smooth one.

Anxiously Awaiting FDDI

When it comes to bandwidth, most network managers have a pretty simple philosophy: More is better. It's no secret that Ethernet and token ring are reaching the saturation point in many applications, and it's also common knowledge that heavily burdened networks are going to be carrying even more traffic. Some of this load will comprise data, but throughput-intensive applications like imaging, video, and multimedia will strain today's 4-, 10-, and 16-Mbit/s nets to the breaking point.

It's no wonder, then, that the industry is eagerly looking to the fiber distributed data interface as the way to ease its current bandwidth crunch—and with good reason. FDDI specifies a transmission rate of 100 Mbit/s, although actual throughput will vary depending upon implementation and choice of protocols. Equally important, its dual token rings will offer unprecedented reliability, while its fiber optic technology will deliver greater immunity to noise and increased security.

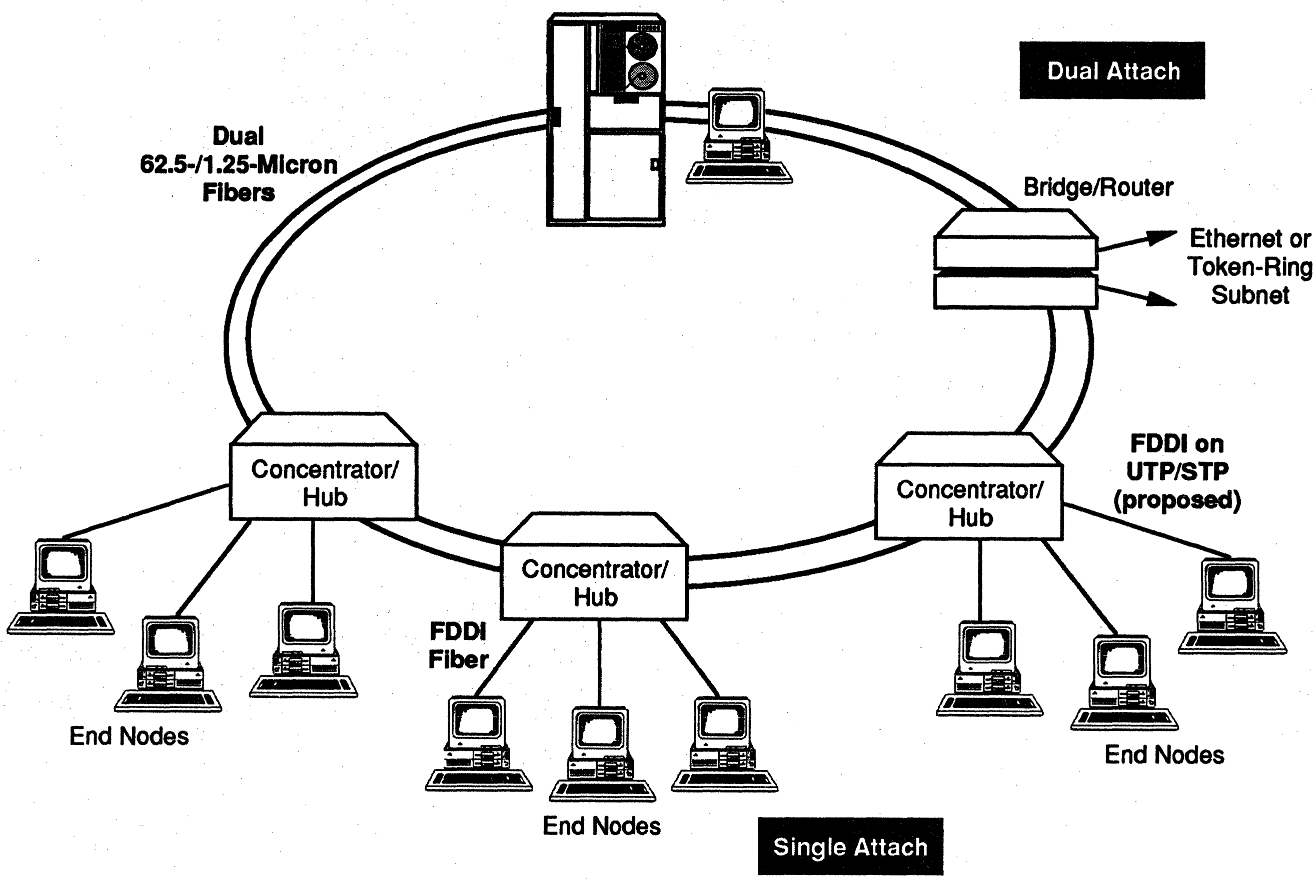
The first step toward getting FDDI to the marketplace, the standards approval process, is drawing to a close. The American National Standards Institute (ANSI) has now published three of the four parts of the full FDDI spec, with the final part expected in early 1992.

As 1992 begins, however, it's apparent that FDDI has entered a second and equally vital stage in which the role of fostering FDDI is being passed from the standards committees to the industry and network managers. In FDDI's second chapter, vendors and users are working to ensure, among other things, that FDDI can handle the demands of today's bandwidth-hungry applications at an affordable price. An effort is also under way to make certain that FDDI products that go into production before the standard is fully approved are interoperable.

Just as with OSI, the first FDDI networks may be priced beyond what some network professionals are prepared to pay. The high cost of FDDI adapters for workstations is one reason. But the prices of these are expected to drop with second- and third-generation hardware implementations. Another reason for FDDI's initial cost is rewiring. Most buildings aren't wired for fiber, and it can be expensive to install—more than $700 to go from the wiring closet to the desktop. Fortunately, a group of vendors is investigating the possibility of building FDDI networks with less-expensive coaxial cable, shielded twisted pair (STP), and unshielded twisted pair (UTP).

This Datapro report is a reprint of "FDDI: Chapter Two" by David Tsao, pp. 59-70 from *Data Communications*, May 1991. Copyright © 1991 by McGraw-Hill. Reprinted with permission.

Figure 1.
FDDI: Doubles and Singles



FDDI–Fiber distributed data interface
STP–Shielded twisted-pair cable
UTO–Unshielded twisted-pair cable

FDDI supports single-connect stations, which attach to one ring, and dual-connect stations, which connect to both rings and can be reconfigured around faults. A wiring concentrator (hub) links the dual ring with single-connect workstations.

The quest for interoperability is being fulfilled by two FDDI test centers. The Interoperability Laboratory (IOL) is located at the University of New Hampshire (Durham); the Advanced Networking Test Center (ANTC) is operated by Advanced Micro Devices Inc. (AMD, Sunnyvale, Calif.). Network managers should query FDDI vendors as to whether their products have passed inspection at one or both centers (see sidebar "Testing for Interoperability").

Like many emerging technologies, FDDI may require tuning and additional refinements to existing software for it to fully deliver on its promise. True, 100 Mbit/s is the figure commonly quoted as FDDI's bandwidth, but it's important to realize that this refers to node-to-node transfers, not system throughput. At the system level, the overhead associated with higher-level protocols like TCP/IP and OSI can slow FDDI.

At the station level, FDDI—like any other networking technology—is at the mercy of its hardware. For example, the network adapter that processes packets during a file transfer has a direct effect on throughput. Similarly, it's unrealistic to expect an FDDI LAN to deliver blazingly fast file transfers if the disk-based subsystem can barely crawl.

The good news is that computer vendors are already hard at work on highspeed protocols that can better exploit FDDI's full bandwidth. And makers of FDDI hardware are bringing to market accelerators and other equipment that will make certain end-users can cash in on FDDI's speed.

FDDI Essentials

In essence, FDDI is a fiber optic, dual token ring that can connect as many as 500 nodes per ring, with up to 1.24 miles (2 kilometers) between nodes and a total LAN circumference of about 62 miles (100 km). The network uses its primary ring for data transmission; the secondary ring can be used to ensure fault tolerance or for data (see Figure 1).

Access to the network is controlled by a rotating token, similar to that passed on a token ring. Unlike conventional token-ring LANs, packets from several stations can share

Testing for Interoperability

Until the FDDI standard is solidified and lengthy field tests are conducted on FDDI components, one of the safest approaches to developing an interoperable FDDI network is to buy components that have been evaluated at either of the two U.S. FDDI test centers.

The Advanced Networking Test Center (ANTC), established in June 1990, is operated by Advanced Micro Devices Inc. (Sunnyvale, Calif.) It has 35 members, each of whom have paid $20,000 for a two-year membership. The Interoperability Laboratory (IOL) was established by the University of New Hampshire (UNH, Durham) to test the equipment used on campus. It has about three dozen members.

Both centers test a broad range of FDDI components, including adapter cards, routers, bridges, concentrators, and intelligent hubs. In addition, both evaluate the performance and interoperability of components in large networks under heavy traffic conditions.

ANTC is a 5,000-square-foot facility wired with multi-mode and single-mode fiber. Shielded twisted pair will be added later. Members who wish to participate must supply system-level equipment that can be hooked directly to the test ring. For example, a vendor with a VMEbus adapter card must provide a complete VMEbus system.

ANTC's primary charter is to test component-level functionality at the PMD, PHY, MAC, and SMT levels:

- PMD includes testing of physical media dependent (PMD) functions such as optical signal rise/fall times, optical signal spectrum, and receiver sensitivity to worst-case fiber, jitter, and degraded signals.
- PHY includes testing of physical layer (PHY) encoding and decoding functions using a broad range of valid and invalid codes.
- MAC tests for media access control (MAC)-level functions, including data transfer, ring recovery, and group addressing.
- SMT includes testing for such station management (SMT) functions as connection management, management information bases, parameter management frames, and timers.

The lab also performs system-level tests such as TCP/IP and X-Windows data transfers. ANTC has found that 80% to 90% of its interoperability problems stem from SMT software implementations. Hardware problems such as buffer management, power supply noise, and link-level monitor errors account for 10% to 20% of the problems. Performance and interoperability problems at the chip level have been less common.

IOL focuses mainly on system-level testing. The majority of the UNH campus is wired using Ethernet subnets (350 nodes), which are linked via an Ethernet. The FDDI network (30 nodes) is located primarily in one building. The equipment, provided by members, includes workstation and PC adapter cards, routers, bridges, and network monitors.

Like ANTC, IOL also performs conformance tests at the PMD, PHY, and MAC levels. However, the main focus is on SMT, which is tested at two levels. First, SMT frame sequences are generated by a test station and transmitted to the station under test to evaluate the equipment's response to maximum/minimum frame size, invalid frames, and so on. Second, multiple stations are connected via fiber to test the physical connection between machines.

Once component-level functionality has been ascertained, the equipment is operated on a fully loaded ring using file transfer protocol (FTP), Telnet, and X-Windows file transfers. To test the robustness of the equipment, parameters such as fiber length between stations are changed, and unusual symbol streams are transmitted onto the network.

To date, IOL has encountered relatively few problems in rings of 30 or fewer stations. Through the end of August, the lab will be extending its tests to much larger rings using equipment from over 40 vendors. These and other such tests will go a long way toward identifying and correcting interoperability problems and increasing the confidence of network managers who are considering FDDI gear.

—By Ken Marrin
President
Davis-Marrin Communications
Oceanside, CA

---

the ring simultaneously. Once a station has the token, it doesn't have to wait for the previous sender to remove the remnants of its packet before it transmits its own packet onto the network.

The links connecting two adjacent FDDI stations consist of two 62.5/125-micron fibers with a full-duplex connector on each end. FDDI can accommodate both single-connect stations, which attach to only one ring, and dual connect stations, which connect to both and can be reconfigured around faults. A wiring concentrator (hub) links the dual ring with single-connect workstations.

FDDI's dual-ring topology and connection management functions establish a fault-recovery mechanism. If a problem arises on the logical ring (such as a broken fiber or malfunctioning station), the primary and secondary rings are collapsed into one, isolating the fault while maintaining a logical path among users. Optical bypass switches enable inactive nodes to pass data directly from one neighbor to another without active power. If a fault occurs in a tree (subnetwork), recovery depends upon the configuration: If the tree is redundant, the logical ring can heal itself; if not, the subnet is isolated.

The Layered Look

FDDI was developed in accordance with the International Organization for Standardization's (ISO's) OSI model. Specifically, FDDI's physical media dependent (PMD) and physical (PHY) layers correspond to OSI Layer 1 (physical); the media access control (MAC) layer corresponds to the lower part of OSI Layer 2 (the data link layer). The station management (SMT) part of FDDI is used to control the other three layers. PMD, PHY, and MAC are published ANSI standards. SMT is about 90% complete and should be finished early this year.

Table 1. FDDI Cost Comparison

	Port	Adapter	Wiring from closet to desktop	
FDDI $4,773-$9,773	$2,000-$3,000	$2,000-$6,000	$773	$110 Fiber $340 Installation $323 Connectors, jumpers
FDDI on STP $3,315-$4,315	$1,000-$1,500	$2,000-$2,500	$315	$157 STP $75 Installation $83 Connectors, terminators
FDDI on UTP $3,000-$3,275	$1,000-$1,500	$2,000-$2,500	$275 for data-grade cable	$180 Datagrade plenum UTP $75 Installation $20 Connectors, terminators
			$143 for voice-grade (DIW) cable	$48 DIW plenum UTP $75 Installation $20 Connectors, terminators

FDDI=Fiber distributed data interface
STP=Shielded twisted-pair cabling
UTP=Unshielded twisted-pair cabling

The PMD layer is the bottom half of the OSI physical layer. It specifies full-duplex connectors, optical transceivers, and bypass switches (optional).

The PHY layer is equivalent to the upper half of the OSI physical layer. In addition to providing 4B/5B encoding, which specifies the electrical transitions that must occur to send data over the media, it specifies a set of line states that perform a handshake between PHYs in adjacent stations. These handshakes are the basis for the connection management protocols used in configuring FDDI's logical distributed topology.

The MAC layer provides peer-to-peer communications for the higher-level logical link control and SMT over the FDDI ring. In addition to token management, MAC handles system timer support, packet framing, and response to normal and system error conditions on the network.

FDDI's basic data-link frame structure is similar to that specified for IEEE 802 frames, with the exception of starting and ending delimiters. The FDDI frame adds a frame control field and frame status indicators. The maximum frame length is 9,000 bytes; data packets can range in size from 128 bytes to 4,500 bytes.

The SMT layer handles such functions as connection management, fault detection and isolation, and ring reconfiguration.

Cost Considerations

At the moment, FDDI's cost/bandwidth ratio is high when compared with mature technologies such as Ethernet and token ring—chiefly because FDDI installations must be rewired for fiber. The cost of running fiber from the wiring closet to the desktop is roughly $730, which includes $110 for the fiber, $340 for installation, and $320 for connectors and jumpers (see Table 1).

Understandably, many network managers are reluctant to rewire their buildings, and soon they may not have to. A more cost-effective approach is to build FDDI networks with coaxial, STP, or UTP cabling. By making FDDI available over existing twisted-pair cabling, the need to rewire for fiber at the station level will be eliminated, as will much of the cost associated with fiber-based node adapters.

The group promoting FDDI over UTP, which includes AT&T, Apple Computer Inc. (Cupertino, Calif.), Crescendo Communications Inc. (Sunnyvale, Calif.), Digital Equipment Corp. (Maynard, Mass.), Fibronics International Inc. (Hyannis, Mass.), and Ungermann-Bass Inc. (Santa Clara, Calif.), expects UTP to reduce node costs from $5,000 or more per port to $2,000 per port. 10Base-T, which is used on Ethernet and token ring, costs about $500 per port. However, at $2,000 per node, FDDI might encounter less resistance in the marketplace, thus boosting sales considerably at the high-volume station level and fueling the overall FDDI market. Of course, the savings of using coax or twisted pair must be weighed against the application's need for security, resistance to electromagnetic interference (EMI), and long-distance transmission. Fiber has a distinct advantage in all of these regards.

For the near term, a compromise solution that combines fiber and copper in the same FDDI network is likely; fiber will be used in the backbone, where its advantages are most important, and copper at the station level, where it is often already installed. The result will be FDDI data rates for much lower cost.

FDDI network components are available but expensive, chiefly because FDDI has yet to benefit from production-level pricing. A VMEbus FDDI adapter card from CMC-Rockwell International (Santa Barbara, Calif.) lists for $11,900, including SMT software and a simple network management protocol (SNMP) driver. In contrast, the vendor's Ethernet card costs from $1,500 to $3,200 (without TCP/IP). For PCs, a single-attachment FDDI adapter card costs about $5,000, compared with about $225 for an Ethernet card.

The TCP/IP protocol stack, widely used in FDDI implementations, is purchased either as source code included in a developer's kit for about $15,000 or preconfigured as binary code for a particular host. Binary pricing depends on a number of factors, including the target machine and the number of installations. SNMP software, which typically runs on the host, is available separately for about $8,000, although it is included with many bridges and routers.

FDDI networks also require a number of other components, including bridges, routers, concentrators, and network monitors. FDDI-to-Ethernet or token ring bridges are available for $15,000 to $30,000, depending on the number of nodes in the network. Routers start at about $20,000 for a single port. An intelligent hub costs from $200 to $500 per port. Prices for a network monitor range from $35,000 to $50,000.

Today, FDDI ports are about 10 times more expensive than Ethernet ports, but many expect this per-node differential to drop to a factor of three by 1993. Ethenet prices will stay about the same, at $500, while FDDI will drop to about $1,500 per port. At that point, FDDI will offer a clear bandwidth/cost advantage over both Ethenet and token ring.

Interoperability

Without a complete standard to guide them, it's possible that FDDI vendors will end up building equipment that is incompatible with offerings from other suppliers. Further, because FDDI is still an emerging technology, most products haven't undergone lengthy field testing, which makes it even more difficult to determine compatibility.

In recent tests performed at the IOL, most interoperability problems increased in larger networks under heavy load conditions, and many were difficult to detect. For example, one semiconductor maker's FDDI chips had trouble dealing with packet fragments (pieces of a packet that the sending station hasn't yet removed from the ring). Under certain load conditions, upstream stations mistakenly interpreted packet fragments as errors. In response, the ring wrapped around the transmitting and receiving stations and performed a bit error rate test (BERT).

Because there was no actual error condition, the BERT did not detect a problem, and the two stations re-entered the ring—at least for the time being. Ultimately, this condition caused stations to periodically drop in and out of the ring. While not catastrophic, it did degrade overall network throughput. (Like most of the bugs encountered at the IOL, this problem has since been corrected.)

ANSI ratification of the PHY and MAC layers has helped ease concerns over interoperability, although the lack of a standard SMT complicates the issue. In fact, many of the interoperability problems detected at the IOL were related to the SMT layer.

Adoption of an ANSI standard for SMT is particularly critical because this layer provides the necessary control that enables all the stations in a ring to work cooperatively with one another. SMT is the mechanism through which stations are initialized and inserted into and removed from the ring. It also provides the means through which data is loaded downline and dumped upline, station statistics are collected, and external authority is exercised over the ring. As noted, SMT also handles configuration management, error detection, and fault isolation—as well as scheduling and address administration.

Fortunately, there is widespread agreement on the bulk of the SMT functions. The remaining questions over the final form of SMT's higher-level, frame-based functions should be resolved by the second quarter of 1992. By that time, FDDI will have been completely standardized. (Two additional FDDI standards—Synchronous Optical Network [Sonet] physical layer mapping and single-mode fiber PMD-are in development. However, neither is expected to compromise interoperability of networks that comply with the MAC, PHY, PMD, and SMT standards.)

System-Level Limitations

No matter how much bandwidth a network supports, node-to-node throughput may be limited by factors that have nothing to do with network performance. Protocol processing, for instance, is a compute-intensive task. Regardless of how fast the network is, a network adapter can only process so many packets per second.

It is important to realize that TCP/IP throughput is independent of the network medium (Ethernet or FDDI) and determined solely by an adapter's performance. Similarly, the speed of disk-based file transfers is directly dependent upon the speed of disk accesses. Right now, the fastest top out at about 8 Mbit/s.

Just how dramatically network adapters and disks can limit FDDI performance was recently demonstrated in two tests at the IOL. The first test involved a two-station FDDI LAN that spanned less than 100 meters. File transfers were made between the two stations employing the user datagram protocol (UDP) for memory-to-memory transfers of raw data. UDP, essentially a connectionless version of TCP/IP, has less functionality than TCP/IP but operates faster. Thus, it is sometimes used for applications in which connection management and reliability aren't as critical.

In this test, FDDI transfers were limited to 33 Mbit/s because the network adapters currently available can only process packets for transmissions at that rate. (In comparison, TCP/IP network adapters for Ethernet can process enough packets per second to support transfers between 1.5 Mbit/s to 6 Mbit/s.)

In the second test, files were transmitted between two nodes using the file transfer protocol (FTP) running on top of a TCP/IP stack. The test included "puts and gets," which consist of a remote logon followed by remote file transfers to or from another station. This time, throughput peaked at 5 Mbit/s to 6 Mbit/s, partly as a result of the additional protocol-processing overhead associated with TCP/IP and FTP and partly because of the need to access hard disks. In this test, actual files were accessed from one system's disk and stored on the other system's; network performance in this instance is thus limited by the speed of the disk subsystem.

The memory-to-memory transfers used in the first test are faster because they eliminate disk accesses on both sides. Since systems generally transfer disk-based files, however, the second test is more indicative of performance in the real world.

Clearly, it's unrealistic to expect high station-to-station throughputs on FDDI LANs saddled with slow disks and network adapters. Higher-speed throughput is achieved with high-performance hardware, such as disk caches and high-speed adapters. This type of equipment is more expensive, and the benefits of using it must be weighed against cost.

CMC-Rockwell International and Fibronics International are both working on VMEbus adapter cards for FDDI; chip maker Protocol Engines Inc. (Santa Barbara, Calif.) is working on an FDDI chip set that includes a dedicated protocol processor. A number of other vendors and silicon houses have FDDI projects in the works but are unwilling to discuss them at present.

Of course, high-speed transfers are easier to achieve when the stations involved are high-end computers, since these can process protocols at a much faster rate and typically have more powerful memory and disk subsystems.

Still, although station-to-station performance is limited by hardware, FDDI does deliver a big performance boost over Ethernet or token-ring LANs because it supports a much larger number of stations before degrading. For example, while Ethernet might support only one or two stations transmitting at 5Mbit/s before degrading, an FDDI network might support as many as 20 stations.

Slow hardware is not the only factor that affects FDDI performance. High-level protocols like TCP/IP and OSI can also take a bite out of FDDI's speed. The overhead for a TCP/IP packet on FDDI, for instance, is 32 bytes, much of which is used for network addresses, error control, and grade-of-service specifications. Over time, that overhead can add up to noticeable performance degradation.

Larger packets may be used to minimize this overhead once a connection is established. Typically, a network protocol establishes its connection using a small packet but adjusts the packet size upward once it determines the receiver can handle them. Sometimes, the protocol uses a default packet size, such as 512 bytes, for data transfers regardless of what the receiver can handle. (TCP/IP is by far the dominant protocol used on FDDI. However, much of what is being said here applies to other high-level protocols as well.)

Not all applications benefit equally from large packets. For computer-aided design (CAD), the network is used primarily to transmit large data files; thus, larger packets can deliver a significant performance boost. On a PC backbone, in which PCs use the network primarily for command-level functions such as remote procedure calls (RPCs), large packets may not help much since an RPC has very little data associated with it. Thus, mostly short packets are transmitted.

The overhead associated with protocol processing and data transmission also affects throughput on the application level. A relatively high percentage of FDDI's bandwidth is accessible when data transfers between nodes are performed at the MAC level—without the use of higher-level network, transport, and file transfer protocols. Fibronics claims it has achieved a throughput of 90 Mbit/s using only MAC-level transfers. MAC-level communications are used chiefly for bridging and LAN interconnection. Almost all other communications, such as peer-to-peer traffic on an FDDI ring, would involve higher-level protocols to some degree.

But as higher-level protocols such as TCP/IP and OSI are added to a network, throughput diminishes significantly. For instance, even with the help of a RISC-based TCP/IP accelerator, CMC-Rockwell was able to reach only 50 Mbit/s when performing TCP/IP memory-to-memory transfers on FDDI networks. It's the overhead associated with processing the TCP/IP protocol that causes the degradation. But Ethernet and token-ring TCP/IP networks usually run at about 1.5 Mbit/s to 6 Mbit/s—so 50 Mbit/s is nothing to sneer at.

One of the most significant sources of TCP/IP overhead is the delay associated with calculating the checksums used to verify the integrity of data. Before the transmitting station can send a packet, it must calculate the packet's checksum and append that checksum to the packet's header. At the other end of the line, the receiving node stores the transmitted checksum, calculates its own checksum, recalls the transmitted checksum from memory, and compares the two.

Another common source of TCP/IP overhead can be traced to the repetitive copying of data that is necessary each time a packet is encapsulated in a new protocol layer. Before data from a user application can be transmitted onto the network, it must pass through several protocol levels, each of which processes it in a certain way. In a TCP/IP network running FTP, for example, the user application (Layers 6 and 7 of the OSI model) sends a file to the FTP layer (OSI Layer 5). The FTP layer, in turn, frames the data in ITP packets and sends it to the TCP layer (OSI Layer 4), which reframes the packet as a TCP packet.

This process is repeated in the Internet layer (OSI Layer 3) and in the MAC layer (OSI Layer 2), which is ultimately responsible for transmitting the packet onto the physical media (OSI Layer 1). The whole process is then run through in reverse when a packet is taken off the network.

The overhead associated with recopying packets through multiple layers depends on several factors, including the speed of the processor, the speed of the memory, the type of network adapter, and the size of the packet.

Packets that are received out of order—a familiar problem in large networks with multiple paths between stations—also add to the overhead. In most TCP/IP implementations, the receiving station monitors the network for a certain range of packets, which are identified by their sequence number. If the packets associated with a particular transmission don't arrive in time, the station won't acknowledge receipt of any of the packets, and all packets within that window must be retransmitted.

Realizing that overhead is a serious problem, many vendors have boosted TCP/IP performance via network adapters with on-board accelerators that speed the functions associated with each protocol layer. To minimize the overhead associated with error checking, for example, CMC-Rockwell uses an AMD 29000 RISC chip and a high-speed DMA (direct memory access) controller (as well as other state machines) to calculate checksums as data is being transmitted, thus eliminating delays. This hardware also offroads 30% to 40% of the host's networking chores, freeing more time to execute application programs.

High-speed network adapters have already boosted TCP/IP throughput on Ethernet LANs from 1.5 Mbit/s to 8 Mbit/s. On FDDI networks, CMC-Rockwell and Fibronics can achieve throughputs of 40 Mbit/s to 50 Mbit/s.

It's important to note, though, that performance on the underlying network doesn't have anything to do with the performance of the protocol stack that runs on top of it. Taking a network adapter designed for TCP/IP on Ethernet that runs at 6 Mbit/s and adapting it to FDDI still yields 6-Mbit/s transmissions.

The XTP Factor

High-speed protocols are also in the works that will be able to take better advantage of FDDI's full bandwidth. One of these is XTP (Express Transfer Protcol), which was developed by Protocol Engines in conjunction with 18 other computer vendors and users. The U.S. Navy has already accepted the protocol as part of its Safenet standard. It also is being submitted to ISO and ANSI committees for national and international ratification.

XTP trades off some of the features of high-level protocols such as TCP/IP and OSI for less overhead, reduced latency (time between acknowledgemnts) in the network, and better performance under steady-state conditions. For example, by carrying a connection request along with data in a packet, XTP eliminates the time-consuming acknowledgment process that occurs when setting up a call. This minimizes connection overhead and reduces network latency by a factor of four. At the application level, this translates into faster execution of commands over the network.

To further enhance XTP's performance, Protocol Engines is working on a chip set to execute the protocol more efficiently. Simulation results indicate that the silicon and protocol combination will be able to hit about 83 Mbit/s at the transport level, thus providing higher station-to-station throughput on the network. The chip set is slated for delivery in 1992 and will cost about $1,000. Interphase Corp. (Dallas) has already announced that it intends to manufacture VMEbus adapter cards based on XTP, and other suppliers are expected to follow suit.

Despite its high performance potential, XTP will face the same lack of installed base and application support that has plagued OSI. To address this problem, Protocol Engines is working on gateways between XTP and other protocols. The first gateway will support application-level transfers through the Berkeley Sockets and Streams interfaces, enabling applications running on top of XTP to communicate with UNIX applications. ■

Multimedia Basics

In this report:

This report will help you to:

- List the basic system hardware and software components on which multimedia is based.
- Determine the costs of acquiring multimedia capability.
- Plan for the use of multimedia in presentations, simulations, education and instruction, and prototypes of new ideas.

If the claims of desktop publishing, CAD and other PC-based revolutions seem to have been exaggerated, take a look at what the movers and shakers in multimedia are saying today about this new "hot" technology:

"It's the convergence of entertainment and computing." —John Sculley, Apple Computer.

"No longer tomorrow's dream. Its time is now." —James Cannavino, IBM.

"Bigger than everything we have today." —Rick Hargrove, Microsoft.

Without question, new desktop computer tools are becoming available to communicators. Called interactive multimedia, or desktop media, this phenomenon has been described as everything from a passing fad to the breakthrough that will make PCs as widespread as televisions and home VCRs.

Basically, multimedia is an intelligent way of combining computer graphics and animation, sound, scanned images and video so the user can extract the most meaning for a particular need. In a more abstract way, it is an object-oriented database consisting of animation, bit-mapped images and audio information that can be accessed in a nonlinear manner.

Given its early strength in desktop publishing, the Apple Macintosh has taken the lead in the multimedia race, but its continued dominance is by no means assured. DOS and OS/2 developers say sarcastically that Apple's most significant accomplishment to date has been in raising the noise level, while third-party Apple developers decry the company's lack

---

This Datapro report is a reprint of "Making Sense of Multimedia" by Bruce Anderson, pp. 32-38, from *Computer Graphics Review,* Vol. V, No. 2, February. Copyright © 1990 by Intertec Publishing. Reprinted with permission.

of clear direction. The fact remains, however, that the Apple Macintosh has spawned a growth industry in multimedia hardware and software, while IBM has scarcely begun to infiltrate the business. (See "What About the PC?" later in this report.)

Multimedia is not really a market, but rather a set of enabling technologies that apply to a variety of markets. Because of this, the R&D investment in the associated hardware and software is forecast to surpass $1 billion annually by 1993. The total market will grow from $400 million to $16 billion in the next five years. With that kind of potential, the technology is clearly going to be part of future computing.

Reaching the Audience

For engineers and architects, multimedia is not a design tool like CAD. Rather, it is a means for educating the audience about a design and persuading them to adopt a particular view. Not surprisingly, early enthusiasm for the technology has come from the field of education. Educators recognize multimedia's role in condensing complex information and making it more understandable, and even fun, to learn. In the professional design market, multimedia will be used most often for presentations to decision-makers, simulations and prototypes of new ideas.

The interactive aspect of multimedia is the crucial component of the new technology, the part that makes a computer essential. Interactive multimedia embraces a variety of existing and emerging technologies. Given the appropriate capabilities, it could ultimately use any of the senses—from sight and sound to smell and touch—in order to immerse the user in the environment intended by the project creator.

Today, however, multimedia is generally based on these specific components:

- *Videodisc:* Although videotape has made the greatest inroads into the video recording marketplace, it does not work well with multimedia because of its linear structure. Searching through a tape for a particular segment can involve lengthy rewinds and fast-forwards, destroying the interactivity of the presentation. On the other hand, a videodisc not only delivers better picture quality, but also has the advantage of fast random-access searching, which is crucial for user interactivity. Several videodisc players on the market can be controlled from personal computers, making it possible to stop and start at precise locations.
- *CD-ROM:* Creating a multimedia presentation of any size will soon use all available hard disk space because color images and sound files are quite large (a single high-resolution color image can easily consume a megabyte or more). Thus, CD-ROM disks are important to the technology, storing as much as 600M bytes on a single platter.
- *Animation software:* Sometimes viewed as a "poor man's video," computer animation has the advantage of being something a nonprofessional can create using clip art, video capture and scanned images. It also allows a designer to work with prototypes and hypothetical designs that have not been developed to the stage where a video could be produced.
- *Speech and music:* For a formal presentation, nothing beats the impact of an audio track, which can include digitized speech, music and other sound effects.
- *Controller software:* Like an orchestra conductor, the controlling software makes the elements of a multimedia presentation work together. The most well-known is Apple's HyperCard software, while on the PC side, similar technology is in the works.

Pricing on multimedia systems made up of these components varies considerably, depending on sophistication (see Table 1).

The Apple Approach

In a major move to establish itself in the forefront of this new wave, Apple Computer kicked off a $15 million promotional campaign last summer called "The World of Apple Desktop Media." Because the company has staked a major portion of its future on multimedia and is farther along in this technology, examining such Mac tools provides an instructive overview of this marketplace.

As Apple likes to point out, the Macintosh has built-in multimedia capabilities. Applications are designed from the ground up to work in a graphics-based environment, allowing picture and

A Case Study in Multimedia on the Mac

An illustration of the trials and triumphs of multimedia comes from Abacus, a small San Francisco graphic arts service bureau that assembled a 12-minute animated Macintosh presentation to help Aerojet, Sacramento, CA, win part of a billion dollar NASA contract.

For months, hundreds of people at Aerojet had been working on a rocket fuel manufacturing proposal, entailing a new fuel-mixing process that required the building of two new plants. As a result actual videotape footage was out of the question for demonstrating the technique. Animation was a last-minute thought . . . and inspiration.

Abacus first prepared a demonstration 40-second animation using Videoworks II, predecessor to what is now MacroMind Director. The preliminary film showed a mockup of the propellant factory: Colored liquids flowed through pipes to a mixing room, where they were blended by a turning screw and poured into an open train car to be hauled away, with the train next disappearing into a tunnel.

When shown to an Aerojet group of about 25 engineers and executives, the presentation received an overwhelming response. Abacus had won approval to go forward with the multimedia project.

With the deadline just nine weeks away, the hard work was just beginning. Fortunately, the script and storyboard were already almost completed. The Aerojet technical illustrator, who provided most of the original drawings, was openly skeptical that a small system like the Macintosh could develop a sophisticated animation presentation, but he was soon caught up in the process.

Abacus upgraded the memory in its machines to 8Mbyte, necessary to contain the animation and sound files. The company developed the essential backup and filing systems and then began the detailed work: Scanning in and cleaning up the original drawings; coloring them in PixelPaint; timing each section from the tape sound-track; animating each of the 26 separate storyboard segments; digitizing the voice track narration; breaking the sound into separate files; animating each segment with Macro-Mind's Accelerator; and finally linking the segments together into a complete show.

A section that contrasted Aerojet's "continuous mix" production process with the "batch mix" method used by its competitor became the critical part of the presentation. To dramatically illustrate the difference, the two processes were shown side by side in the animation. Here, Videoworks' power to process up to 24 moving objects simultaneously proved invaluable, although an intense effort was required to get this complex portion running properly.

At a preview for the Aerojet proposal team, the full-blown production was also clearly a success. Project management decided, however, that because of the lengthy pauses required to load each of the 26 segments into memory (averaging 3- to 4Mbyte apiece), the whole program should be transferred from disk to videotape for the final presentation.

Thus began a new set of problems. The initial results of the transfer and editing steps were terrible. Thin lines vibrated up and down on the screen. Graphics close to the edge were cropped off, and bright colors bled into other color areas. Five days of additional work were required to fix the problems, followed by yet another videotape transfer.

Finally, the production was delivered—after consuming 1,300 work-hours by three full-time computer artists and half a dozen free-lancers. It had required some 80Mbyte of disk space, using 700 floppy disks for backup. (The company has since installed a tape drive.)

A few months later, word came that Aerojet had won the contract. The multimedia presentation had played a significant role in the victory.

---

sound elements to be readily cut and pasted between programs. Audio capabilities, including speech and music synthesis using the MIDI (Musical Instrument Digital Interface) standard, are built into every machine, with stereo sound included in higher-end Mac models. And, perhaps most important, HyperCard software is shipped with every Macintosh.

HyperCard is a natural environment for interactivity. By adding a series of visible or invisible

Table 1. Multimedia Setup Costs (Macintosh-based systems)

Monochrome System:	
• Macintosh SE, 20Mbyte hard disk, 1Mbyte RAM	$3,800
• Animation software	500
• Sound digitizer	300
• Hand-held scanner	500
• LCD projection screen	1,600
Total:	(about) $7,000
Color System:	
• Macintosh IIcx, 40Mbyte hard disk, 4Mbyte RAM	$7,400
• 19-inch color monitor and board	6,500
• Animation and paint software	1,000
• CD ROM player	1,000
• Color scanner	7,000
Total:	(about) $23,000
Video System:	
• Macintosh IIci, 80Mbyte hard disk, 8Mbyte RAM	$10,700
• 19-inch color monitor and board	6,500
• Color video camera	1,500
• Videotape player	1,000
• Video recorder	3,000+
• Video digitizer board	2,500
• Animation and video software	4,000
Total:	(about) $29,000

Depending on individual budgets and multimedia needs, setup charges for a Macintosh-based multimedia system can range from about $7,000 to almost $30,000.

"buttons" (areas on the screen the user can select with the mouse pointer), the multimedia programmer encourages users to follow their own paths through the information, going deeper into areas of particular interest. Graphics and sound capabilities are an integral part of HyperCard. Through the use of external program routines called XCMDs, HyperCard can control videodisc players, CD-ROM drives and other multimedia peripherals.

Programming of HyperCard is straightforward, using a simple language called HyperTalk. In effect, the software allows control of a rich database, in which graphics, text, animation, video and sound are merely elements linked by the programmer to produce the desired result.

If HyperCard is too limiting, more power is available through third-party software selling for less than $500, including SuperCard, from Silicon Beach Software, San Diego, and Plus, from Olduvai Software, South Miami, FL. High-resolution color, large-screen display, more extensive programming features and stand-alone run-time output are just a few of the enhancements these products bring to the field.

Beyond HyperCard, the Macintosh strategy in multimedia today is animation; programs based on interactive text or still color pictures look weak by comparison. The nucleus of Macintosh animation is The MacroMind Director ($695) from MacroMind, San Francisco. This program represents a major update to the company's pacesetting Videoworks program that has been used for virtually all Macintosh animation presentations in the past, including the disk-based, guided tours on how to use the Mac that Apple ships with every system.

Because animation is a collection of single frames flashed quickly on the screen, the developers at MacroMind employ a spreadsheet analogy to enable users to lay down pictures and sound one frame at a time. The sheet reads from left to right, as in a music score, and the columns hold video and sound activity that will take place as each frame is displayed. Up to 24 channels may be controlled at once, each representing a different event or animation element, resulting in complex sequences of independent motion and audio.

Standard animation tools, such as wipes and dissolves, moving titles, etc., are built into Director, along with a limited auto-animation technique called "tweening." With this capability, a programmer can indicate the beginning and ending positions of an object, along with the number of frames wanted to accomplish movement, and Director will interpolate the transformation for smooth animation.

Multimedia also goes beyond using MacroMind Director with other Macintosh enhancement products. Obtaining images can be done via a color scanner or plug-in video frame grabber board. Because the technology is still in its infancy, the boards are expensive, typically $2,500 and up. But the link between computer and video can only become stronger.

Audio capabilities can be enhanced through use of a sound digitizer such as the $249 MacRecorder from Farallon, Berkeley, CA, the most widely used package of this type. The software allows recording from a microphone or other source, in mono or stereo, along with digital sound editing and compression of the disk file up to 8:1.

What about the PC?

IBM might have been caught napping when the desktop publishing revolution began its march, but the company now intends to be in the forefront of the multimedia push. Although Apple has initially attracted the usual bevy of experimenters and creative types, IBM is focusing on its traditional turf: large corporations, government and institutions.

Since 1986, IBM has been covering the industrial multimedia market with its InfoWindow system ($4,395), which uses a videodisc controller as the centerpiece, along with a touch screen and a pair of speakers incorporated into an EGA color monitor. It has widespread use in the corporate training environment and supports a catalog of hundreds of commercially available courses.

IBM has also announced hardware and software directly positioned against Apple's offerings, hoping to increase its stakes in multimedia. The software is built around the Audio Visual Connection ($495), an authoring language and application development tool to synchronize sound, pictures and text under DOS 4.0 and OS/2. On the hardware side, video capture and high-quality audio boards, priced at $2,250 and $565, respectively, are now available for the PS/2 line of computers.

But full-motion video requires approximately 10Mbyte per second of storage space, making digital compression a must. So for the future, IBM is concentrating on Digital Video Interactive (DVI), which promises 100:1 video compression for efficient use of storage. Working with Microsoft and Intel, IBM appears committed to bringing full-motion video to the computer screen, with the ability to store 70 minutes of video on a single compact disk.

Such presentations will primarily appeal to high-end markets, however. They need first to be processed on Intel's Application Development Platform ($21,500), an 80386 machine specially configured with seven plug-in boards. This provides a low-resolution image, compressed at 40:1, that is then sent to Intel for further compression at $250 per minute. Finally, the resulting digital output is forwarded to a CD-ROM manufacturer for creation of a disk master.

IBM plans also include an "integrated multimedia PC," which is expected to combine a 386-based computer, 2Mbyte of RAM, VGA graphics, a digital sound processor and a CD-ROM drive, all for under $3,000. This may steal some of Apple's thunder.

Microsoft, has launched a new Multimedia Division to create software extensions and application programming interfaces that will support the addition of stereo sound and synchronized video to the Windows and OS/2 Presentation Manager operating environments. End-user products are expected to follow in two or three years.

The fight between Apple and IBM over multimedia will last several years, as both explore the technology's true potential. And in the wings, the Japanese are preparing major entries into the multimedia battle as well.

Animation, video and sound can be incorporated into HyperCard, although animation must be accomplished by quickly flipping cards as in the turn-of-the-century penny movie machines. MacroMind provides an Accelerator to make Director output run more smoothly inside HyperCard, and also an Interactive Toolkit that allows HyperCard stacks and other multimedia programs to run inside Director.

Although most serious multimedia programmers require a package like Director to integrate their presentations, a CAD user who is creating simple 3-D visualizations or prototypes might find all the animation capabilities necessary in more basic graphics packages, including Super 3D from Silicon Beach, Swivel 3D from Paracomp, San Francisco, and Dimensions Presenter from Visual Information, La Puente, CA. Creating a design a CAD program or directly in the 3-D package, an architect or engineer can create walk-throughs, rotations, camera movements, etc., without using a multimedia package. For smooth animation, narration and interactivity, these files can generally be exported to Director as well.

A Demanding Technology

Wonderful as interactive multimedia may be, it places significant demands on both the hardware and the person creating presentations. CD-ROM devices, for example, supply the storage space necessary for complex material, yet their access times and data transfer rates are significantly slower than hard disks (averaging 500ms vs. 28ms), making smooth animation and sound difficult to achieve.

The computer itself requires a fast, powerful CPU and large memory capacity (4M bytes or more) to keep up with simultaneous audio, video and animation channels while polling the interface device to check for user input. When it comes to real-time video, today's technology leaves much to be desired. Because current Macintosh and PC hardware cannot move information fast enough through the computer bus, users must compensate with a slower video rate, smaller picture size or fewer colors.

On the creative side, it is a fact of life that the easier a process may be for the user, the more difficult it becomes for the creator. The Macintosh itself is an illustration of this; programmers struggle mightily to produce the intuitive interface that allows simple "point and click" operation.

In a multimedia presentation, the creator must do a significantly greater amount of work to make the result seem simple and communicate the message. Video production, for instance, is very expensive ($2,000 per minute and up) and requires a good deal of expertise. A generation raised on professional films and TV shows is not going to be impressed by a home movie look.

For this reason, the education and training fields have been the first to embrace multimedia technology. Because they repeat information to a succession of new recipients, productions tend to have a longer life-span, making it worthwhile to invest a great deal at the "front end" of a production.

ABC News Interactive, for example, has introduced the first in a series of HyperCard-based multimedia packages, "Vote '89: The Campaign for the White House." Incorporating existing video footage from the presidential campaign and instructional slides, the program is aimed at students studying the various issues and conflicts of the campaign. By clicking the mouse pointer on a rocket icon, for example, the student can discover a candidate's stand on defense issues.

One attempt to go beyond the merely instructional approach is being developed at the Lawrence Berkeley Laboratory's Center for Building Science, Berkeley. Researchers there have created a prototype building envelope design tool using HyperCard and a videodisc player. Text, graphics, color video, sound and animation are combined to provide a multimedia database of building case studies, including walk-through architectural models and lighting design analysis. The ultimate goal is to create an electronic, interactive master reference, enhanced by an artificial intelligence overlay to provide expert "advice" on implementing the data in the design process.

An engineer creating a multimedia presentation for a single prototype demonstration, on the other hand, will be less likely to invest the time and effort necessary for a full-scale showing. There is also the problem of sources. Although a graphic designer can create an impressive illustration on one of the desktop art packages, an engineer generally must look to clip art and scanners for the raw materials of an effective presentation.

In larger corporations, coordination may be required with the company's communications or audio-visual department to create an accurate yet professional-looking presentation. Unquestionably, multimedia technology favors a team approach.

Despite the resources required, however, multimedia is a technology that is certain to grow more prominent in the coming years. With Apple, IBM and Microsoft funding the development, the packaging of information will become more refined as the tools are made easier to handle.

As usual, competitive pressures will determine how quickly the technology makes inroads. Similarly to CAD environments a few years ago, a company only needs to lose one or two bids to a slick multimedia presentation to be convinced of its importance.

In the meantime, despite all the claims, multimedia technology stays ahead of the understanding of its potential users. But isn't that the way all previous computer revolutions have hit? ■

Microcomputing in the Year 2000

In this report:

Datapro Summary

During the past decade, the personal computer has dramatically changed the way in which we work. PC technology has placed computing power into the hands of the most pedestrian of users. As we drive to the year 2000, rapid enhancements in semiconductor, mass storage, and other technologies makes the future almost boundless. The challenge for the computer industry will be to produce new, broad-ranging applications and markets.

Popular forecasters of the future tend to believe that anything they imagine can be implemented shortly thereafter. Part of the blame for this expansive state of mind rests with the computer community. Indeed we have accomplished many wonders in the past 40 years, and it appears additional wonders are *technically* feasible by the year 2000.

During these four decades, however, we also researched tunnel diodes, perceptrons, wire memory, and many other technologies that somehow got sidetracked. One of my early articles for *Computer* magazine (in 1977) reflected the then bright glow of magnetic bubble memory. A couple of years later I basked briefly in the early morning warmth of optical storage, a technology still struggling for a foothold in the computer market. Another example: The original version of the Next computer employed on optical disk, but the second model retreated to magnetic disk.

So, the future is out there, but it is obscured by a great deal of fog. Well, let us press on.

Some technologies have proved to be surprisingly successful. The semiconductor march promises to continue for another decade. Intel chairman Gordon Moore said as much in an IEEE Computer Society Compcon keynote address in 1989. Consequently, all those facets of information technology that rest in part on semiconductor technology—processors, memory, digital signal processing, and communications—should continue to drop in price and increase in performance.

The progress of magnetic recording has not been prompted by a well-known "law," but it has been equally remarkable. Other technologies have also advanced rapidly and new technologies continue to appear. This technological progress will result in an enormous expansion in computing capability. More capability can support more ambitious applications, but these have to be found.

Still, we must remember that semiconductor and magnetic-recording technology are the exceptions among the technologies. Both have advanced rapidly for a long time. Some technologies contributing to information products are not so lucky. Some have advanced slowly—or fitfully. Other have lost out because they did not manage to catch up with existing technologies.

The Laws of the Semiconductor

The advance of semiconductor technology has been so consistent for so many decades that industry seers concocted "laws" to characterize past progress and to project the future. Here is a sampling of laws from Moore:

- The number of transistors per chip doubles every two years (a law originally handed down in 1975 and still going strong).
- Feature size has been cut in half every six years.
- The amount of charge needed to distinguish between two states has been reduced from five million to 100,000 electrons—a factor of two each generation.

An 11-member expert panel offers a somewhat different version:

> Logic and memory component density is expected to increase by a factor of 10 every five years through the end of the century.[1]

(The panel was established by the Aeronautics and Space Engineering Board of the National Research Council to project the information and computer technologies that would be possible by 2000 to support aeronautics if sufficient resources were available.)

Now, in the beginning of the last decade of the second millennium, the effect of these and other laws have brought us to 4-Mbit memory chips and one-million-transistor microprocessor chips. If the laws continue to hold, we should reach 256-Mbit memory chips and 50-million-transistor microprocessor chips by 2000. Researchers to not expect to reach the ultimate limits set by the laws of physics during this decade.

Memory

Manufacturers began producing the 4-Mbit chip in large volume last fall.

> Quadrupling the capacity of each generation of DRAMs [dynamic RAMs] has been achieved by reducing cell size to one third and increasing chip area by one half.—Fujio Masuoka

The next generation, 16-Mbit, is being developed by many manufacturers. Hitachi researchers have announced the fabrication of the following generation—64 Mbit—in the laboratory, but yields are still extremely low. The researchers are using electron-beam technology with a feature size of about 0.33 µm. Volume production of the 64-Mbit chips is said to be about five years off. In the meantime, research is in progress on a new way of applying light waves called phase shifting. This method may extend optical lithography down to the 0.33-µm level, or even lower. With X-ray lithography, researchers can reduce feature size to 0.20 micrometers, making the 256-Mbit chip possible, perhaps by 2000. Table 1 summarizes this density progression.

Problems, such as supply voltage and cell capacitance, turn up all along this projected trail, though researchers expect to overcome them. Masuoka[2] most recently surveyed these problems.

Logic Chips

In microprocessors, Intel's 80486 and Motorola's 68040—each with more than one million transistors—represent the current top of the line. Intel plans to introduce the 80586 in 1992, according to executive vice president Craig Barrett. One chip will hold four to five million transistors.

An 80486-based motherboard by Pioneer Computer demonstrates the current state of the personal computer art. In addition to its microprocessor and eight slots for optional boards, it can house up to 256 Kbytes of secondary cache memory, up to 32 Mbytes of main memory, and a math coprocessor.

Bruce Patterson, Intel's managing director in Australia, speaking at a conference in Sydney last June, looked still farther out to the 22-million-transistor 686 by 1996 and the 100-million-transistor 786 by 2000. The expansion of microprocessor capability is outlined in Table 2.

> Over the last 30 years, the MIPS/dollar ratio of computers has increased by a factor of over one million.—Patrick G. Gelsinger, Paolo A. Gargini, Gerhard H. Parker, and Albert Y.C. Yu[3]

By 2000, Gelsinger and his group at Intel project 50-million transistors on a one-inch-square die operating at 250 Mhz. They assume four parallel processors on this chip, operating to 750 MIPS (million instructions per second) each. In addition, transistors would be available for floating-point operation, data and instruction caches, vector units, a graphics unit, and various controllers.

Moreover, an interface better suited to human characteristics, involving speech synthesis, voice recognition, motion video, and image recognition could be accommodated on this size of transistor budget. This budget could also accommodate fault-tolerant design, redundancy, built-in self-test, and error detection and correction.

Wafer-Scale Integration

The one-inch-square chip that Intel anticipates is about four times the area of today's largest chips. In the meantime, research continues in the US, Japan, and Europe on advancing to wafer-scale products. In Japan, progress is ahead of schedule. Tadashi Sasaki reported to the 1989 International Conference on Wafer-Scale Integration. He expects to see this technology in use before 2000.

"This much is certain," Sasaki added. "New process technology oriented toward wafer-scale integration will be

Table 1. Projected Schedule of Increases in Memory Density Per Chip

Memory Density (Mbits)	Feature Size (µm)	Type Radiation	Commercially Available
4	0.80	Visible light	1990
16	0.50	Visible light	1992
64	0.33	Electron beam or ultraviolet	1995
256	0.20	X ray	2000
1,024	0.10	X ray	?

Table 2. Projected Growth of Microprocessor Power

Designation	No. of Transistors (millions)	Commercially Available	Clock Rate (MHz)
486	1.2	1990	40
586	4.5	1992	60
686	22.0	1996	100
786	50-100.0	2000	250

introduced gradually into chip design, as designers find new ways to resolve the problems."

Digital Signal Processing

A DSP is basically a special-purpose processor designed for exceptionally fast execution of certain instructions (particularly multiplication and addition). The multiply-accumulate combination occurs in many algorithms employed in processing real-time signals. Thus, processing signals coming in from the analog world via an analog-to-digital converter demand this kind of computing power. Similar digital processing has to be performed before the results are sent out via a digital-to-analog converter to the real world.

> Between 1976 and 1986 a three-order-of-magnitude reduction in cost, size, weight, and power consumption occurred.—L. Robert Morris[4]

> Between 1986 and 1988 the state of the art advanced from fixed-point to floating-point architectures, allowing an increase in dynamic range and precision.—Stephen A. Dyer and L. Robert Morris[5]

Since the DSP is essentially a type of microprocessor on a chip, it will benefit during the 1990s from increasing density. A likely development is that the DSP function will migrate to the same chip that holds the other functions necessary to the operation of a complete computer system. The signal processing involved in such activities as speech recognition, speech synthesis, three-dimensional moving graphics, image and video processing, and data compression and reconstitution will no doubt occur on this one multifunctional chip. With the data remaining on this chip during several kinds of processing, speed can be greatly increased.

In the past, it has been impractical to integrate both the analog circuitry involved in going from the analog realm to the digital realm (and vice versa) and digital processing circuitry. Digital switching "pumps" the chip's ground level to a degree that interferes with the precision of the A/D or D/A conversion processes. Recently, however, a new conversion technology moves the operations requiring precision into the digital realm. This technology promises to make integration of A/D and D/A conversion feasible on this ever-expanding chip.[6]

Edward A. Lee[7] believes that DSPs may become the key to the much touted—but little realized—integration of telecommunications and computation.

Communications Chips

Caught up in the excitement of the greatly increased bandwidth provided by fiber optics, we sometimes lose sight of the fact that, from another vantage point, a communications network may be thought of as a vast collection of computers, connected by transmission links. Those computers, or switches as the telephone companies call them, are also based on semiconductor technology. Since the fiber optic links will have enormous transmission capacity, they will be capable of carrying telephone voice, teleconferencing, data, images, television, home shopping, and services not yet conceived.

Consequently, the computer switches will need much more capacity, too. The same argument applies to local area networks and metropolitan area networks. One of the capabilities that will be incorporated on these switching chips will be the logic to interface to the high-performance network.

Most users will be hooked up to networks and use this interface logic. Most likely even if you are a hermit holed up in a cave with a personal computer, your computer will contain a chip with all these interface capabilities. It will be less expensive to include what most users want on a 50-million-transistor chip than to produce a separate line of chips for hermits.

Neural Networks

This technology is just beginning to be implemented on chips. So far most neural-network applications have been simulated by programming on conventional digital computers. This approach enabled experimenters to learn something about applications that neural networks could accomplish, but performance, of course, was very slow.

In the last few years several laboratories constructed neural networks in hardware. Carver Mead, Michael Emmerling, and Massimo Sivilotti at the California Institute of Technology fabricated 22 "neurons" on a chip. The neurons are actually a feedback network of operational amplifiers. At present, several companies market simple neural-network chips. For example, Intel's chip features 64 neurons connected to each other and to 64 inputs through 8,192 synapses.

One expects that the hardware embodiments of neural networks will increase in density during the 1990s, along with other types of processors implemented on silicon. Hardware versions should execute thousands of times faster than conventional simulations.

There are challenges ahead, however. One challenge is the steep learning curve demanded of development engineers who implement neural-network applications. In the early years of microprocessors, Robert N. Noyce, one of the founders of Intel, once remarked that the manuals sold better than the chips! Engineers had to learn how to use them. Neural networks will go through that stage, too, John J. Hopfield of Caltech told me in an interview.

A second challenge is the number of connections between neuronlike elements. "In the typical VLSI [very large scale integration] circuit, the output current of a typical transistor feeds two or three other transistors and receives its input from a similar number," Hopfield noted.[8] In neurobiology, that number is on the scale of 3,000 rather than 3. [which is] a qualitative difference."

Of course, the number of connections between neurons in current artificial neural networks is nowhere near the number of connections in biological neural networks. Even so, say researchers at Arizona State University, "the area required to route connections and to contain the average length of interconnections increases at unacceptable rates as more processing elements are added."[9]

A third challenge is to reproduce the learning capability of a biological network in an artificial neural network. Presently, artificial networks "learn" by setting the strengths of interconnections between the operational amplifiers. This learning occurs in a separate phase distinct from operations "by running through a massive database of information on the problem under consideration," Hopfield points out.[10] Biological systems learn on the fly while also continuing to perform. They also require much less information for learning than artificial networks.

Finally, current artificial neural networks are very simple compared with natural networks. The biological brain includes many capabilities that neurobiologists have not yet sorted out. Sorting them out is both difficult and time-consuming. The task won't be finished by 2000.

Dark Horses

Meanwhile, at least four dark horses in the research barn—or just out of—are preparing to challenge silicon technology. The farthest out in front is gallium arsenide (GaAs) since relatively small digital chips have been commercially available for five or six years. Computing with superconducting, optical, or molecular elements is still in the research stage.

GaAs

GaAs technology offers "clock rates between five and 10 times higher than CMOS [complementary metal-oxide semiconductor] capabilities," according to Ron Gates.[11] However, he said GMOS using less power "will remain the clear choice for most digital circuits." That still leaves the high-performance end of the marketplace for GaAs.

At the board level, GaAs devices work in conjunction with silicon chips to implement circuit paths in which propagation speed is critical. "In 1985, a commercially available GaAs circuit containing a few hundred transistors was rare," Cates said.[12] In 1990, "GaAs integrated circuits with over 100,000 transistors are mass produced and found in mainstream computer and telecommunications systems."

> Continuing improvements in GaAs process techniques indicate a doubling of device density every nine to 10 months.—Cates

He expects microprocessors fabricated in GaAs to appear in the early 1990s. For some years to come, then, if the marketplace welcomes them, GaAs chips may follow a growth law comparable to the silicon pattern.

Superconducting

Two enticing lights appear at the end of this tunnel. First, superconducting Josephson junctions switch about 1,000 times faster than silicon transistors. Second, the heat dissipation of superconducting circuits is very low, which permits them to be packed tightly, reducing transmission time between circuits. This, "superconductors might be used to develop computers that would operate a million times faster than today's fastest computers, " according to a 1990 report of the National Commission on Superconductivity, established by the US Congress in 1988.[13]

Of course, there are several serious scientific difficulties between here and there. That is why superconducting for computer applications remains in the research stage and promises to stay there during the decade. A handful of Japanese researchers continued work in the field after IBM withdrew in 1983. AT&T continued a modest effort. Now both the National Commission on Superconductivity and a study sponsored by the US Defense Dept. Advanced Research Projects Agency have recommended that the US make a greater effort.

A group at the Hitachi Central Research Laboratory believes that "all of the techniques essential for a large-scale computer will be developed within several years." For the superconducting computer to become a practical product will require breakthroughs in circuitry, powering, and packaging. Thus, the Hitachi group concluded: "The Josephson computer might appear in the 21st century."[14]

Optical Computing

Photons and electrons are little "somethings" that move at high speed and carry information. Electrons currently serve as the basic element in digital computers. Many scientists believe that photons can be harnessed for the same purpose. In fact, photons have been embodied in synthetic-aperture radar and other forms of optical signal processing. These kinds of processing, however, are not digital.

In another application, photons convey digtial information over fiber optic links. The switches between these links are presently digital electronic, involving frequent conversions between photons and electrons. Establishing some optical switches at the linkage points would be more efficient.

Perhaps that is one reason why AT&T Bell Labs is one of the most active players in optical computing research. In January 1990 its research group, led by Alan Huang, demonstrated a prototype digital optical computer. Lasers generated the photons and optical on-off switches controlled them.

IBM, for one, has its doubts. "In the early and middle 1960s, researchers at IBM and elsewhere explored the potential of digital computing with photons instead of electrons." Trudy Bell reported,[15] "However, the digital techniques seemed impractical because of limitations in the physics and technology of materials and the devices that apparently could be built from them."

Of course, there have been significant developments since the 1960s, but the IBM researchers are still skeptical. "They feel that the problems encountered in the 1960s are basic to the physics involved, not merely the technology," Bell said.

A recent survey by the *New York Times* concluded that "scientists believe that optical on-and-off switches and even optical switchboards will one day replace their electronic counterparts in computers, although there is general agreement that practical [digital] optical computers are still many years away."[16]

Molecular Computing

We have the existence proof in the form of organic life that protein molecules perform various processing operations analogous to computing. For example, when foreign organic material enters the body, certain molecules recognize it as foreign and initiate a process that rejects it. Presumably a string of appropriately designed molecules in some artificial array could sense some phenomenon, "compute" what it means, and start other molecules working to do something about it.

Considerable biological research is underway all around the world and some research directed at this area commenced in 1983, according to Micahel Conrad.[17] "It is likely that in the next several years some molecule devices will be exhibited that perform primitive information processing," he said. Biosensors are likely to be the first molecular application.

Of course, we need to learn much more about how life forms function, how to abstract out some very simple process from the complexity of evolutionary life, and how to construct the protein molecules to implement this process. "Marketable products seem decades away," Conrad concluded.

Mass Storage

The race between mass-storage technologies continues. A decade ago observers thought that optical storage would take over some part of mass-storage applications because of its high bit density. On a different tack, as knowledge of

Moore's law on density spread, chip enthusiasts looked forward to a time when some part of the data traditionally held in mass storage would migrate to main memory. As it happened, however, during the 1980s magnetic density kept pace with semiconductor density and gained on optical density.

> Capacity of magnetic recording quadruples every three years (the same rate at which the capacity of dynamic random-access memory is increasing).—Roger Wood[17]

> Magnetic storage density has been increasing by a factor of four every five years, a trend expected to continue.—NRC expert panel

Magnetic Recording

The parameters that determine recording capacity include linear bits per mm, areal bits per sq mm, tracks per (radial) mm, data rate in Mbits per second, magnetic properties of the magnetic medium and the recording or reading head, signal processing techniques, and so on.

Let us look at one of Wood's[18] parameters—areal density. IBM's 1989 high-capacity 3390 magnetic rigid disk records at 96,000 bits per sq mm. The IBM Magnetic Recording Institute (San Jose, Calif.) demonstrated a density of 1.8 Million bits per sq mm. "Some design challenges remain," Wood noted. "IBM has yet to announce commercial plans for the technology in a disk drive."

In spite of a history dating back to 1900, however, Wood sees no danger of stagnation in areal density advancements. "There do not appear to be any physical limits to prevent stretching densities to many megabits per square millimeter," he said.

Optical Storage

Erasable optical systems currently record in the range of 400,000 to 700,000 bits per sq mm. Read-only compact disks record near 1 million bits per sq mm. The drawback to wider use of optical disks is not storage density. The difficulties in many products are relatively high cost, slow access time, and lack of erasability (or write capability). On the plus side optical disks are immune to head crashes and the disk is removable. Hence a great volume of data is separately stored or carried between computers.

> Optical storage density is expected to increase at a rate slightly slower than that for magnetic technology.—NCR expert panel

Optical mass storage is a prime example of a technology with great promise that unfortunately began life in the wake of a continually developing technology (magnetic). Always a step or two ahead of the trailing technology, magnetic recording generated the necessary funds from expanding sales to continue its exponential rate of improvement.

Semiconductor Storage

The areal density of a 4-Mbit dynamic RAM is 1.2 million bits per sq mm. Of course, future generations will have still greater density. Putting eight or nine of these 4-Mbit DRAMS on a card provides storage capacity equivalent to a 4-Mbyte hard disk, but with the direct-access advantage of semiconductor architecture. These semiconductor "disks" should become more competitive with hard disks during the 1990s as DRAM density increases and prices decline.

Intel's new 2-Mbit Flash Memory devices illustrate the principle that when internal feature size drops, external dimensions decline, too. At 20 mm x 8 mm x 1.2 mm, a flash memory device can be mounted on each side of a memory card and still fit within the 3.3-mm thickness permitted by the governing standard.)

Of course, all three of these mass-storage technologies—magnetic, optical, and semiconductor—are "moving targets." As we contemplate an increase in the storage capacity based on one technology, we need to keep in mind that the other two technologies are also moving forward. In each case, a host of factors in addition to areal density—cost, reliability, performance, form factor, no moving parts, power consumption, and so on—will pay a role.

The People Interface

At present, the input to a microcomputer is overwhelmingly via keyboard; the output is a CRT display about half the size of a sheet of paper. The reason, of course, is that these two pathways to people are the least expensive. People have other ways of putting out information besides a keyboard—handwriting, diagrams, and voice. They also receive information in other ways, particularly speech and other sound signals. Moreover, their visual field is much larger than the small screen. So they might like to view a much larger screen.

Now all of these modalities are already in use, usually in a small way. More widespread use has been delayed by the need for additional hardware to accommodate them. That would raise prices. In addition, the algorithms to implement some of these interfaces—particularly handwriting recognition, voice recognition, and speech synthesis—are not yet entirely worked out. Moreover, it is to clear if people feel an acute need for these additional paths. At least, they have not surged eagerly to acquire the interface capabilities already on the market. Perhaps they believe the interfaces are too expensive for what they do.

The exponential growth of hardware capability, founded on the semiconductor laws, promises to provide the computing capability needed to implement these interfaces during this decade—at a cost much of the market will find bearable. it appears that much greater computing and memory capacity will be necessary to accommodate algorithms that adequate handwriting recognition, voice recognition, and speech synthesis will require. The neural-network principle is expected to do well at these tasks. The marketplace will decide whether people really want these added capabilities.

Handwriting Recognition

Hand-printed capital letters (or other carefully constructed characters or symbols, perhaps in prescribed boxes on a form displayed on an entry tablet) can be recognized now. Several commerical products are on the market, with several more expected in the next year or two, including Windows H from Microsoft. Recognizing unconstrained cursive writing, Japanese and Chinese characters, and handwritten mathematical equations is much more difficult.

Professional writers who think that current word-processing technology puts too much machinery between their muse and their manifest words might adapt better to cursive-writing input. Just how they create the words seems to be a delicate subject for some creative writers. In particular, they might welcome the opportunity to edit a manuscript with a stylus, crossing out one word and writing in another in the traditional way. They would like to view more text than the small screen presently provides.

Similarly, mathematicians might seize upon a notebook computer that permits them to write a variety of symbols, superscripts, and subscripts in their usual manner.

Moreover, many professionals like to make a point with a diagram. Certainly one could use a stylus to draw a diagram and store it in pixel form. How it could be further processed and for what purpose remain to be invented.

The drawing, storing, and processing of kanji characters is an analogous problem. Naturally, this greatly interests the Far East.

All these difficult tasks are mainly a matter of developing complex algorithms. Consequently, predicting the progress made by 2000 for such efforts is more chancy than to forecast increased transistor density. At least the hardware capability to implement the new algorithms inexpensively will become available during the decade.

Voice Recognition

Inputting information to a computer by voice would be even easier than by handwriting. The capability has been available, at least in rudimentary form, since the mid-1970s. I remember reporting on such a system at that time. Aircraft maintenance was one of the applications. The mechanic, both hands at work, audibly reported part numbers and conditions to a central computer. The system accepted only a limited vocabulary, required each spoken word to be separated, and had to be trained to each user.

This system was a far cry from a secretary to whom you could casually direct a comment in colloquial speech. Performance has improved since then, "comfortable and natural communication in a general setting (no constraints on what you can say and how you say it) is beyond us for now, posing a problem too difficult to solve," said Richard D. Peacocke and Daryl H. Graf of Bell-Northern Research.[19] "It seems likely that we will also need natural language understanding before we can achieve comfortable and natural communication with computers through voice."

So, again, we have to know more about the patterns of speech on the semantic level. It will take complex algorithms to decipher them, which will likely require a lot of computer power. In the meantime, relatively simple applications will continue to grow.

Speech Synthesis

Text-to-speech conversion is now in common use in reading devices for the blind and in voice mail systems. However, it has not taken the personal-computer world by storm. Very likely sighted persons would just as soon, under routine circumstances, read messages as hear them.

"Our current understanding does permit the synthesis of intelligible speech," Michael O'Malley[20] wrote recently, "but our models, especially in their dynamic behavior, are not yet adequate to make synthetic speech that is indistinguishable from human speech." "Generally speaking, text-to-speech systems are limited by our current knowledge of linguistics," he continued. "The resulting speech sounds somewhat boring, but most attempts to make it more lively result in some sentences having a foolish-sounding prosody."

As to the future, O'Malley said that "unrealistic expectations for dramatic future improvement . . . sometimes arise from an unsophisticated view of the complex linguistic information involved. Improvement will continue at the same slow, steady pace that has produced incremental progress in accuracy, intelligibility, and naturalness over the past three decades. However, we can expect faster progress in methods of delivering the technology."

By 2000, much more powerful hardware will accommodate more sophisticated algorithms at a lower price. The growth of speech synthesis is likely to revolve around finding applications that people find useful.

Displays

What is the wish list in this area? It includes color, brightness, higher resolution, double-page screens with windows capabilities, capability to move images and three-dimensional drawings, viewing angle, reponse time, light weight, low power, ruggedness, portability, and minimization of radiation. Most of these attributes are already available—at a price. During the next 10 years, because of advances in hardware, the price per attribute is certain to decline.

CRTs will continue as the display of choice over the next few years, according to a recent report by the Office of Technology Assessment (OTA).[21] In the longer term, however, the Office expects a shift away from CRT to technologies such as liquid crystal, electroluminescent, or plasma displays. Less well-known display technologies are also under investigation.

"By the late 1990s, yet another display technology may become available—the active matrix flat panel liquid crystal display," OTA speculated. Ten- to 14-inch color displays of 640 x 400 pixels have been demonstrated, mostly by Japanese companies.

> Display size (diagonal) is increasing at the rate of three inches (75mm) per year.—OTA

Commercial production is about three years behind research and development demonstrations. High-definition television, when it takes off, will create the market to support a large-screen, high-resolution display. It will then be available for computer applications, too.

Making It Happen

Outlining where the semiconductor laws and other advances might take us by 2000 was the easy part. We, the human race—or at least some part of it—need to make it happen.

Maintaining the pace of technology development during the next 10 years will take huge investments. While the immediate funds may seem to come from investors, banks, and company profits, they come from the marketplace, directly or indirectly in the ultimate analysis. Products have to be sold to large markets and the proceeds reinvested in development.

A growing market for computer products requires new applications, additional users, or more intensive use by existing users. Where are the mass-usage applications comparable to word processing or spreadsheets? Where are the new users? How can we encourage present users to do more?

The answers to these questions generally involve software. Some answers may lie in integrated systems, easy-to-use interfaces, open systems, and standards. If appropriate communities of manufacturers, marketers, and users can agree on these matters, software developers can produce the implementing software. But developers cannot program consistent people-computer interfaces, for example, if they cannot agree on what they are.

Now, all of this involves people seeing the possible ways to accommodate the semiconductor evolution, agreeing what these ways should be, and then learning to use them. That takes funds. Moreover, it takes time—usually time measured in years. A relatively small engineering group may develop the next-generation microprocessor quite rapidly. Getting people to adopt the resulting paradigms is a slower process. Yet it takes funding from a large number of people to support the development groups and new fabrication facilities.

Huge Investment

The money necessary to keep the semiconductor industry on its exponential growth curve has also been increasing on its own exponential path. Twenty-three years ago Intel developed two new technologies—silicon-gate MOS and Schottky bipolar—for about $2 million. Moore remembered. Today, the cost of developing a new technology is around $60 million. Intel recently announced that its capital budget for 1991 is $600 million. The cost of building a fabrication facility is in the $200- to $400-million range. Later in the decade, because of the increasing complexity of denser processes, that facility cost may climb to $500 million. An X-ray process in the late 1990s may reach one billion dollars.

Even large firms are finding it necessary to join forces to develop next-generation technologies. The other side of these costs is that recouping the investment requires heavy product sales. To put it another way, companies must justify the funds necessary to finance the next generation.

> The number of transistors produced doubles every year.—Moore

This is approximately what all these exponential curves mean in terms of market expansion. The production curve is currently going through about four million transistors per person in the industrialized world.

> Manufacturing cost per chip will begin to increase as feature size penetrates the fractional micron range, but cost per unit capability will continue to decline.—NCR expert panel

What is the industry going to do to sell transistors in these incredible quantities and at higher chip prices? Well, one outlet is new applications.

New Applications

Just three application programs account for the bulk of personal computer software: word processing, spreadsheet, and database management. Tens of thousands of application programs divide the rest of the sales among themselves. An estimated 40,000 programs run on the 8086. Perhaps desktop publishing is the most recent big application area to arrive on the scene. Another is laser printing; it takes a lot of transistors to print in many fonts.

"If it doesn't do something that the people are doing right now in great frequency, it couldn't possibly be that important." Charles Simonyi of Microsoft Corp. declared in a September 1991 theme issue of *Byte* largely devoted to the next 15 years. Well, one could amend that statement to say "what people *want* to do right now," but the computer power and programs are not yet available.

> We've seen the price per instruction per second, which dropped at a rate of roughly 15% a year for the previous two decades, drop by more than 50% over the last several years,—Stephen C. Johnson[22]

One activity in which people do engage is television watching. Combining television with a personal computer may result in an application area that absorbs some of those millions of transistors. This combination already has a name: multimedia.

Multimedia has been available from Intel, for instance, in the form of add-on boards. Originally the seven add-on boards cost about $22,000. In February 1990, Intel reduced the system—which it calls digital video interactive (DVI)— to two boards for $2,000. Then last November Intel released a pair of micrprocessor chips (750) with DVI for about $100. DVI enables personal computers to present full-motion video, high-resolution still photographs, animation, and stereophonic sound. More than 200 companies are developing products based on this set. The price of a multimedia-equipped personal computer—expected to cost an additional $1,000—will soon be within the range of many consumers.

Storing multimedia digital data will require either enormous quantities of mass storage or high-speed data compression and reconstitution. The latter solution will use more millions of transistors.

Another promising application goes by the name of groupware. As defined by Terry Winograd[23] of Stanford University, groupware is "a new kind of software that supports cooperative work." Other participants in this new field call it coordination theory, collaboration technology, or computer-supported cooperative work. For the most part, activity in this field is still at the research stage—trying to figure out the mechanics of how people work together in groups. Still, a dozen or so ideas have been reduced to actual software and some are for sale commercially.

For example, an electronic meeting system developed at the University of Arizona with IBM support is intended to get more ideas out in the open. In hierarchical organizations, lower-ranking participants are often loathe to express their opinions freely. The electronic meeting system permits them to interject their ideas anonymously via keyboard.

Some of the best minds in the software business are trying to find sizable new applications. Thousands of application programs are jostling for position in today's marketplace, with only a few becoming commanding applications. No doubt one or two more sizable applications will turn up by 2000.

More Intensive Use

Since uncovering market-expanding software applications is tough, the next recourse is to enlarge the market by encouraging present users to use their computers for additional tasks. Niche programs provide reasons for more intensive use.

However, practical obstacles slow this expansion process. One obstacle is the lack of standardization between program interfaces. That is, everything from function-key actions to little procedures differs from one program to another. That makes it hard for people to use as many different programs as their work might justify.

Some developers have tried to make program interfaces more similar to each other, that is, develop "integrated" systems of several programs. Various forces have hampered these attempts, including product differentiation, rapid release of new generations, and copyrighting of "look and feel."

Electronic mail is an example of an application that could be used much more extensively. Everyone who wants a substitute for telephone tag is a potential customer for it. Just about everyone has a telephone, but the estimated number of people who actually send messages from their home terminals at least once a week is only in the hundreds of thousands.

At the present, several factors are slowing the growth of e-mail. First, it is hard to use. The complex procedures to address a recipient are even more complex if that recipient is on another network. There is no universal directory of e-mail addresses. And—a chicken-or-egg problem—relatively few of one's correspondents use a e-mail. Ideally e-mail should work as easily as the telephone system. Parenthetically, it took the US telephone system 40 or 50 years to move from scores of independent systems to one integrated system.

Another obstacle is the conflict between proprietary systems and open systems. Some companies see an advantage in locking in their customers with proprietary software. From the standpoint of growing the overall market, however, open systems seem to be more effective.

Integrated systems rest on a structure of standards. Building that structure takes time.

> The cost-performance or size/performance of algorithmically specialized functions such as signal processing, database management, and flow model processing will be improved by the year 2000 by factors of two to three orders of magnitude, primarily through the use of parallel and distributed architectures.—NRC expert panel

Expand the Market

The bulk of computer systems sale today takes place in the industrial world, mainly the United States, western Europe, and Japan. One billion consumers live in these countries, and not all of them have yet been reached. More than four billion people make up the developing world. In time they will absorb great quantities of transistors.

Time

People and organizations resist change. Some constitutionally resistant. Others are willing to change, but the process itself takes time.

Therefore, the technology development outlined in this report may proceed more slowly than technically possible. Shortage of funds may slow progress. The shortage may result form an inability to increase the market enough to generate the necessary funds to develop technologies. The market growth may be slower than hoped for because great new applications do not get discovered. In any case most of the people and organizations may decide to advance at a slower pace than the technology could theoretically allow.

In nature, exponential growth curves always top out. They turn into S-shaped curves. The fundamental reason is that the expanding species outstrips the environmental resources needed to support the next increment of expansion. For the same reason the rapid, exponential expansion of the computer industry will slow down when it outstrips its resources, or whenever the market fails to supply exponentially increasing resources.

I know you hoped I was going to lay out the next 10 years concretely—so you could plan on it! Instead I have offered glimpses of a much booby-trapped obstacle course, littered with conundrums about what people and their organizations may do.

The technical challenges ahead of us in this decade are as great as they ever were in past decades. The "end of history"—a thought that was briefly popular a year or two ago—is not yet. The technological future is almost boundless, the "laws" tell us, certainly for this decade. Many organizations that understand these possibilities and are willing to finance further investment, seek broad-ranging applications, open new markets, and exploit present markets more intensively will grow and prosper.

References

[1]S.F. Lundstrom and R.L. Larsen, "Computer and Information Technology in the Year 2000—a Projection," *Computer*, Vol 18, No. 9, Sept 1985, pp. 68-79.

[2]F. Masuoka, "Are You Ready for Next-Generation Dynamic RAM Chips?" *IEEE Spectrum*, Vol. 27, No. 11, Nov. 1990, pp. 110-112.

[3]P.P. Gelsinger et al., "Microprocessors Circa 2000," *IEEE Spectrum*, Vol. 26, No. 10, Oct. 1989, pp. 43-47.

[4]L.R. Morris, "Guest Editor's Introduction to Digital Signal Processing Microprocessors," *IEEE Micro*, Vol. 6, No. 6, Dec. 1986, pp. 6-8.

[5]S.A. Dyer and L.R. Morris, "Guest Editors' Introduction to Floating-Point Digital Signal Processing Chips," *IEEE Micro*, Vol. 8, No. 6, Dec. 1988, pp. 10-12.

[6]H. Davis, R. Fine, and D. Regimbal, "Merging Data Converters and DSPs for Mixed-Signal Processors," *IEEE Micro*, Vol. 10, No. 5, Oct. 1990, pp. 17-27.

[7]E.A. Lee, "Programmable DSPs: A Brief Overview," *IEEE Micro*, Vol. 10, No. 5, Oct. 1990, pp. 14-16.

[8]J.J. Hopfield, "Artificial Neural Networks," *IEEE Circuits and Devices*, Vol. 4, No. 5, Sept. 1988, pp. 3-10.

[9]M. Walker, P. Hasler, and L. Akers, "A CMOS Neural Network for Pattern Association," *IEEE Micro*, Vol. 9, No. 5, Oct. 1989, pp. 68-74.

[10]W. Myers, "Artificial Neural Networks Are Coming," *IEEE Expert*, Vol. 5, No. 2, Apr. 1990, pp. 3-6.

[11]R. Cates, "Gallium Arsenide Finds a New Niche," *IEEE Spectrum*, Vol. 27, No. 11, Apr. 1190, pp. 25-28.

[12]R. Cates, M.J. Helix, and K.V. Rosseau, review of "Gallium Arsenide Digital Integrated Circuit Design," *IEEE Spectrum*, Vol. 27, No. 11, Nov. 1990, pp. 11-12.

[13]A. Pollack, "Making Computers a Million Times Faster," *New York Times*, Sept. 26, 1990, p. C7.

[14]Y. Hitano et al., "A 4-Bit, 250-MIPS Processor Using Josephson Technology," *IEEE Micro*, Vol. 10, No. 2, Apr. 1990, pp. 40-55.

[15]T.E. Bell, "Optical Computing: A Field in Flux," *IEEE Spectrum*, Vol. 23, No. 8, Aug. 1986, pp. 34-57.

[16]M.W. Browne, "Magic Crystals: Key to New Technologies," *New York Times*, Feb. 20, 1990.

[17]M. Conrad, "The Lure of Molecular Computing," *IEEE Spectrum*, Vol. 27, No. 5, Oct. 1986, pp. 55-60.

[18]R. Wood, "Magnetic Megabits," *IEEE Spectrum*, Vol. 27, No. 5, May 1990, pp. 32-38.

[19]R.D. Peacocke and D.H. Graf, "An Introduction to Speech and Speaker Recognition," *Computer*, Vol. 23, No. 8, Aug. 1990, pp. 26-33.

[20]M.H. O'Malley, "Text-to-Speech Conversion Technology," *Computer*, Vol. 23, No. 8, Aug. 1990, pp. 17-23.

[21]Office of Technology Assessment, *The Big Picture: HDTV and High-Resolution Systems*, Government Printing Office, Washington, D.C., June 1990.

[22]S.C. Johnson, "Hot Chips and Soggy Software," *IEEE Micro*, Vol. 10, No. 1, Feb. 1990, pp. 23-26.

[23]T. Winograd, "Groupware: The Next Wave or Just Another Advertising Slogan?" *Digest of Papers*, IEEE Computer Society Compcon, Los Alamitos, Calif., Spring 1989, pp. 198-200. ■

Future Trends in Personal Computers

In this report:

Datapro Summary

In the past 10 years, the personal computer has impacted nearly every aspect of our lives from recreation and entertainment to education and business. Experts predict that the PC and computer technology in general will undergo revolutionary changes in the future. Advances in molecular, optical, and superconducting systems are expected to take us to the next threshold of personal computing.

The Future Personal Computer

Today computers are everywhere, embedded in television sets, microwaves, automobiles, portable bibles, electronic personal calendars and telephone directories, electronic spelling checkers, electronic dictionaries, language translators (often with voice output), mobile FAX/phone/data terminals, and so forth. It is the age of the information appliance. Already the typical household has more computers than motors. This trend is likely to continue, with still more fantastic computer-based appliances on the way. But nothing is likely to change more dramatically than the personal computer. The PC is expected to be quite different in the year 2000. The following projections are based on many of the ideas coming out of a prediction contest conducted by Apple in the mid-1980s that resulted in a concept known as the electronic notebook.[1]

This Datapro report is a reprint of Chapter 5, "Future Trends and Projects," pp. 101-113, from *Business Decisions with Computers* by Daniel Schutzer.

Physical Description

The future personal computer will look different physically from today's models (see Figure 1). It will typically come in three sizes and speeds.

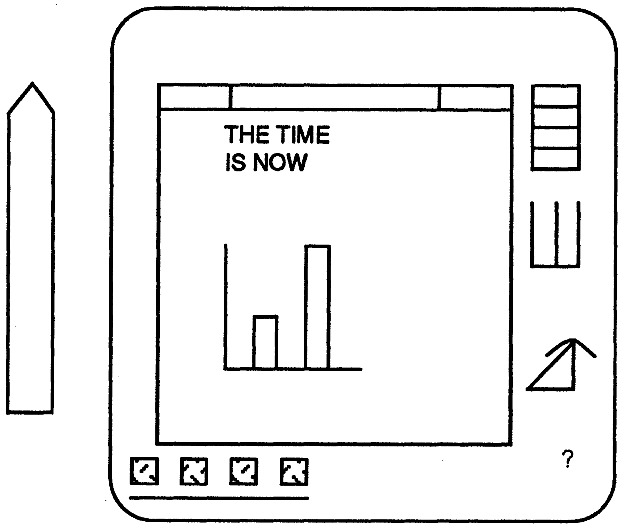
The largest machine will be about 1¼ feet by 1 foot by 1½ feet deep. It will fold in half to form a portable package. The next largest will be about half that size, and the smallest is expected to be the size of a pocket diary. It will function as a diary, communications device, and notebook.

The display will quite likely cover the entire face of the computer. It will serve as both an input and an output device for the majority of applications. The screen is flexible, pliant, and touch-sensitive (able to tell which areas of the screen are currently depressed and how much pressure is being used). It produces high-resolution (probably 4098 $\times$ 4098 pixels) color images of high brightness, but uses low power. The screen's hardness can be modified electrorheologically by electric current to give the screen the ability to mimic the feel of many substances; e.g., the screen keyboard can mimic the IBM keyboard click.

The display will probably include a feature for time-lapse video overlay to handle video telephone or CCTV (closed circuit television) monitoring features.

Advances in matrix printing and input scanners will allow the coated screen to be used as both a scanner and a printer. You can lay a paper image or photograph on the screen and it will be scanned and captured.

Figure 1.
The Personal Computer of the Future



If you lay a blank page over the image on the screen, the image can be transferred to the paper.

There is a central processor and two or more standard co-processors (e.g., a 160-bit floating-point processor and an interactive string processor), with slots for optional co-processors. The processor will probably be about 20 MIPS, upgradable at the top end to about 200 MIPS (supercomputers will be in the terraops range in about 15 to 20 years). There is a minimum of 48 megabytes of memory, with room for expansion to 256 or more. The circuits are divided over both sides, which are connected by a flexible ribbon across the hinge, so the computer can be folded in half.

Communications ports are standard and most likely optical. They will be capable of handling I/O (input and output). The PC communicates with the outside world to send and receive data, books, faxes, school lessons, and movies loaded into the computer, or processing assistance from remote specialized processors, or to rent and access storage space from third-party vendors, with any associate billing electronically transferred. The fax transmission will employ a fractal-based data compression technique to squash bit-mapped images several thousandfold and allow them to be included as "literals" in a vector description of a picture.

Since the communications ports are optical, there is no need for a physical link to the outside world. It is sufficient to rest the computer on a desk-based cradle for connection. There will also be zone phone points where telecommunications can be established just by standing within a certain distance of a physical marker.

The wide area network port will include a single-chip ISDN (Integrated Services Data Network that includes voice, data, graphics, video, and TV transmission) handler capable of running two channels at prevailing rates (currently 64 Kbps), and one channel operating as fast as 64 to 100 Mbps, perhaps greater (there already exist 32 Mbps optical networks).

The communications port gives access to both a local external device bus and a wide area ISDN network. The local area fiber optic network can run at speeds from 100 Mbits to 1 gigabit. The local port also can connect to peripheral devices such as printers, two 500 megabyte smart cards (they can store data on the card), and a single co-processor bus slot for third-party application-specific processor chips.

The PC will also include a data encryption co-processor for both secure communications and efficient data storage, and a speech analog/digital processor for both speech recognition and speech generation that can be output directly, or via the ISDN for voice messaging.

User Interface

The future PC is expected to have an intuitive, customizable user interface that includes speech, touch, handwriting recognition, and seamless communication with the external world, both local and wide area, and data, voice, and image. All communication will take place transparently; there are no special commands or procedures for the user to learn. The interface will have a high-level graphic description language for use as a communications protocol for sending messages, data, and faxes over an ISDN line.

The interface will be pictorial and intuitive, with all the services integrated together (METAPHOR, or Hewlett Packard's New Wave). The user will interface with the computer through a graphic-window-based interface. Objects represent tasks, files, applications, background tasks or services and a "soft" keyboard. The interface will include a "natural" integration of all forms of input: keystrokes, mouse-derived pointing, handwriting, and gestures to perform functions. The display changes and adapts to the function being performed. For example, when entering text, a QWERTY keyboard is displayed. When working on a spreadsheet, the display rotates 90 degrees into a landscape mode with a wide spreadsheet displayed, with a numeric keypad that can move around the spreadsheet, and a handwriting note box to allow the entry of smaller amounts of text information. For paint packages, paint pots and stylus brush are provided. For music, there is representation of the musical instrument and a place for writing notes on musical staves on the screen. Each application can have its own unique input and output display.

Because of this application-unique display and input, there will be greater use of application-specific buttons; e.g., for a text editor, there are "save," "load," "font," "style," and application-specific "help" buttons. To retrieve a document, you will only need to point to its symbol or write the new document name on top of a currently accessed document. You can send a document or an application to another user by dragging its icon onto the icon of a telephone.

You will also be able to create your own handwritten shorthand symbols for tasks. To delete a file you may just scribble over it, or select it for removal by drawing a circle around it, or double-click a mouse pointed to it. To merge mail, you hit the form letter, write a plus symbol anywhere on the surface, tap the relevant database symbol, and then draw an arrow toward the required output.

Input and output sound will also likely be present (the NEXT computer is an example of this). Sound, when used appropriately, can be very effective in helping to reduce the clutter on the display. Sound also provides meaningful clues, such as signaling the arrival of electronic mail. Speech commands are useful to start jobs to run in the background. Assume the screen is occupied with database forms. You start the process to collect your electronic mail, or to look up relevant news stories coming in over the news wires. The computer acknowledges the command with an audible signal and a new active task icon. When the mail is collected, a bell may sound and an icon flash or change color to indicate its arrival.

The ability to customize will be a major feature of the user interface because different people have different styles. Some people like pictorial icons (you will even be able to design your own); others are happier with words. The system will let you place function keys wherever you wish—on top of the keyboard, or to the side. Furthermore, the system will remember your preferences, even if they relate to a remote application.

By putting input and output into the same device, we have for the first time the possibility for truly intuitive interaction with the application. With a spreadsheet, you can slide the numeric keypad anywhere on the screen. You can play chess and move the pieces with your finger.

Although many people use multiple windows on a screen to represent various multitasking activities, most applications and people work best on a full screen. Therefore, the page metaphor is likely to become very popular. With the page metaphor, you change applications like turning pages in a book. There will be simple screen-based buttons to page backward or forward as a standard feature. For cut and paste jobs, or for times when you want to sit back and watch your various computer domains working away on their own, windows, both overlapping and tiled, will be available. When you size a window you will have the option of viewing a portion of the whole application or seeing the full screen scaled down.

User Scenario

Consider the following scenario: When you turn on your computer, you will see a sheet of electronic paper. You sign your name across it, or speak a command. Your signature or your voiceprint is your password. The system matches your characteristic patterns of pressure, speed, and shape, or sound. Since it is 9 A.M., the time when you normally start work, the system makes the safe bet that it should bring up the document you were revising yesterday when you stopped. The displayed document is a page of printed text covered with handwritten marks and jottings where you have indicated changes. It might even be possible to grab the top page of the screen with your fingers and scrunch it into a ball when you want to abandon a badly written document, and then place it on the wastepaper basket.

Systems Software

The multitasking operating system is resident in 4 megabytes of ROM (read-only memory). Most of the needed functions will be built into the operating system, reducing the applications programming to little more than a sequence of operating systems calls with a user interface. The core of the system modules includes memory management, scheduler, message dispatcher, communications controller, and several service engines, each performing a particular type of generic task. They include a calculating engine for floating-point math, an editing engine, a dialogue engine to support user interface, a typography engine, a drawing engine, a searching engine, a sorting engine, and a filing engine. There could even be a logical (or rule-based) inferencing engine. Flame technology has already announced a Parallel Inference Machine (PIM) that comes as a chip which can be attached to existing computers and workstations. The PIM is capable of 1 million LIPS (logical inferences per second), which is equal to about 1000 instructions/second.

In many cases these engines make use of an underlying hardware co-processor. The functioning of the operating system, however, will not depend on any particular co-processor being available. If a suitable co-processor exists, the operating system will use the module in preference to its own software routines. The operating system will use a software bus to string these engines together in different configurations to solve specific problems. It will be object-oriented; engines will appear as "black boxes" joined by channels along which they exchange encapsulated objects and messages. Application programs will share the same software bus as the operating system, and some will work entirely through operating system's functions. We will be able to treat any application as a new engine for building more applications.

Besides its more conventional functions, the operating system controls the architecture, monitoring the various processors and managing the local power supplied to them in an effort to preserve power. Accordingly, the computer is a "soft" machine. There is an on switch that starts the machine by connecting its local power source (most likely a thin film battery) to the main circuit, but powering down is a software function. There is a secure reset button that can be activated by the machine's own coded stylus. The local power source will be good for at least 12 hours of continuous operation, with the capability to recharge in 10 minutes from any external power source.

Application Programming

Programming for the user will be by example (or, if preferred, by flowcharts). When programming by example, the user steps through the job sequence in learn mode and then saves the sequence as an intelligent agent which performs the customized task when called upon. The agent may be generalized by asking you to enter data and parameters at run time. It will work like a combination keyboard macro-pipe-batch file-job control program to the 100th power. Because of its seamless communications capability, the computer can use agents on remote computers as well as its own local ones.

The system will contain a full-scale object-oriented database engine whose file structures are designed to facilitate hypertext-style cross-referencing between the contents of different documents. Large amounts of indexing will often be performed on the fly. For example, as you type text, the operating system will examine each word after you press the space bar. It will look it up and consider it a candidate for indexing against already defined categories for cross-reference. And when you load data in batch mode, the operating system will automatically do string functions like searching for categories and spell checking.

Uses

With such a computer, the possibilities are almost limitless! It could be used as a help in selling. For example, you don't just get a picture of a video recorder on the screen, you get a three-dimensional working image of the recorder. You can touch the buttons and operate the recorder before you buy it. Your working environment will resemble the world around you. Electronic banking will present you with a check to sign that will look like a real check and require a real signature. In database work, you will use forms and charts that more closely resemble the old-fashioned hand-data entry, and sketches rather than the rigorous fields and graphs of the current computer environment. The possibilities seem to be limited only by the individual imagination.

More Extreme Predictions

There are those who predict a radically different future for the personal computer. They predict that the ultimate personal computer will not look like an electronic notebook at all, but rather will be the size of a golfball. We will communicate with it through speech, and it will respond to us with voice outputs, and where it is useful, with projected holographic images.

Still more strange, the computer might be a set of computer-controlled goggles, headphones, and gloves, or possibly even a body suit (or data suit), that send the user off into an artificial reality where he or she can become part of a scene from a movie, a travelog, or some surreal fantasy world. Such a system would have the potential to transform entertainment, education, engineering, science, medicine, and even pornography.[2] Crude artificial reality machines already exist in the university and military laboratories to train pilots in artificial cockpits, and to allow architects to design a building and then to lead a client on a tour of it. There is even a commercial product based on the artificial reality concept. Mattel, Inc., introduced a $90 computer glove that enables users to control some Nintendo games with hand gestures. The glove is partly based on the design of a glove built by VPL Research, Inc., an artificial reality firm.

Of course, artificial reality might become too appealing, like an electronic LSD, and people might seek to ban it. In fact, Timothy Leary, the former Harvard researcher who popularized LSD in the 1960s, claims that artificial reality is "getting closer and closer to the psychedelic experience."

Revolutionary Technologies

In recent years, we have seen dramatic increases in performance being achieved with the silicon semiconductor computer. Semiconductors keep halving in linear dimension. The result is a doubling of the speed and a quadrupling of the capacity. Advances with this technology can probably continue only four or five more times before reliability and yields become unacceptably low. Then where will we get the future dramatic improvements from?

Research attention is now being directed toward developing new computer architectures and materials that will further improve speed and performance and mimic human intelligence. Some of the technologies researchers are investigating include parallel processing, including neural networks; superconducting chips and transistors; optics; and biological material. All these projects were included in the newest budget proposals submitted by Japanese government-industry consortiums to MITI, the Ministry of Trade, which is known for funding key research thrusts. There is also the possibility of more exotic breakthroughs like superconductor computers and molecular computers.

Superconducting

Advantages of Superconducting

Until the late 1970s, the density of circuits in the state-of-the-art silicon chip had been doubling each year. Since then, the advances have slowed to a doubling every two years. Higher circuit densities also boost the electrical resistance of the chip. Chips with more electrical resistance run hotter, increasing the chances of failure. So as the density of the silicon chips have increased, the need for cooling has increased. Today, some 10 percent of the cost of installing a typical mainframe goes into buying the air conditioning system. In fact, today's supercomputer systems are often dwarfed by their companion cooling systems. If a supercomputer that has no heat energy loss could be built, it could have its circuitry more densely packed, resulting in shorter propagation times and faster processing speeds. Superconducting offers this potential.

Recent Advances in Superconductivity

Superconductivity was first identified by the Dutch physicist Heike Onnes in 1911. Until a few years ago, the effect was observed only when certain metals were chilled with costly liquid helium to nearly absolute zero, or -459°F. In the early 1980s, two scientists at IBM's Zurich Research Laboratory, K. Alex Muller and J. Georg Bednoz, began looking at ceramic oxides. Ceramic oxides are typically poor conductors, actually insulators, at room temperature. In April 1986, they reported that a ceramic achieved superconductivity at 13°F above the highest temperature achieved for the best metal alloy.

Paul Chu made history in February 1988 when his team of physicists at the University of Houston and the University of Alabama chemically combined four elements to create a compound capable of superconducting at the record -283°F. This is above the boiling point of liquid hydrogen, which is -320°F. Liquid hydrogen is a cheap and easily obtained coolant. The chemical was dubbed 1-2-3 by IBM scientist Grant and his colleagues. The name refers to the atomic ratio of its components: 1 part yttrium, 2 parts barium, 3 parts copper, and a varying amount of oxygen. Soon after rival researchers at the University of California at Berkeley discovered the compound, and since then it has spread like wildfire. Other firms with active research programs in superconductivity include GE, AT&T, IBM, Hewlett-Packard, Dupont, Lockheed, Westinghouse, University of Houston, University of Alabama, University of California—Berkeley, Stanford, Cornell, MIT, University of Illinois, Department of Defense, Department of Energy, NEC, NT&T, Toshiba, and Fujitsu. Japan has put together a consortium on superconductivity that is similar to its fifth-generation computer project.

Of course, the stable, high-temperature superconductors discovered to date would require even more expensive cooling systems than the current crop of supercomputers. But a cost-effective supercomputer built using superconductivity could probably be achieved in steps.

On-Chip Connections

Practical problems have been discovered with superconductivity. IBM research indicates that some of the new high-temperature superconductors actually conduct electricity at a slower rate than copper. The current flows through copper at one-fourth the speed of light. Another problem limits the potential of superconductive interconnects when they are used between conventional Integrated Circuits (ICs). As ICs heat up, they could raise the temperature of interconnects enough to lose their superconductivity. On-chip connections could be limited by the same problem. But replacing today's metallic on-chip connections with supercomputers might offer a bigger offsetting gain. They could significantly reduce the chip's power needs, while allowing denser circuit packing. Thin-film fabrication of the 1-2-3 substance was demonstrated at Stanford in March 1987, and at IBM in April 1987.

Josephson Junctions

A more distant prospect for the use of superconductors in computers would be the development of ICs on which even active elements are fabricated from superconductors. One candidate would be Josephson junction devices.

Josephson junctions are superconductor-based switching devices that operate 50 times faster than today's best silicon transistors and 10 times faster than the best semiconductor devices, which are made of gallium arsenide. In addition, they have a very low power consumption and a very low heat dissipation. Problems remain with realizing the practical implementation of superconducting Josephson systems. However, none of the problems appears insurmountable.

Superconducting Computers: A Reality?

An all-superconductive computer is probably at least a decade off. If it comes, it will probably advance in steps. The Nobel Prize-winning physicist and computer scientist Kenneth G. Wilson, currently at Cornell, adds: "Superconductive computers will have to go all the way to mass production before they can be said to have any significant impact on the computing world. As far as I can tell that's a long way away."[3] William J. Gallagher, manager of applied cryogenics at IBM's T.J. Watson Research Center in Yorktown Heights, New York, who is overseeing some of IBM's superconductivity research, says: "It's all very much blue sky. But it's somewhat less blue sky than it was last month and an awful lot less than it was a year or so ago."[3]

Even if we never do see a superconducting computer, all this focused research is already laying the foundation for another sort of major impact. Materials science has no well-developed modeling; it is still very much in the province of the professional simulation. So the biggest breakthrough of all this activity may not be a computer at all, but a breakthrough in analyzing and modeling materials science.

Optical Computers

Advantages of Optical Computers

Optical computers hold the potential to reach speeds a thousand times faster than the most powerful conventional machines.[4,5] Light moves faster than electrons through semiconductors; in fact, an optical computer could operate at speeds reaching the physical limit.

Light can transmit a lot more information than electricity. This is what has prompted the use of optical fibers for communications in place of digital electronic transmission. Additionally, an optical computer would be more efficient for routing telephone calls transmitted over fiber optic lines because it would not have to convert back and forth from electricity to light.

Because photons do not interact, light beams can cross each other without interference. Therefore optical computers could be more interconnected than their electronic counterparts. The increased interconnection means we could design computers with shorter paths, and the length of the path determines the speed of the computer.

This property would also make it easier to implement parallel processing on an optical computer, and to solve problems with large amounts of data by accessing lots of data in parallel. Indeed, future optical computers might be a solid 3D block, allowing for still more interconnections.

An optical computer would also be more flexible than an electronic computer, since its circuitry would consist of beams of light that could be reconfigured more easily than wires.

The military is interested in optical computers because they are harder to break into and would not be disabled by the electromagnetic burst resulting from a nuclear blast.

Recent Advances in Optical Computers

On January 29, 1990, Alan Huang, a researcher at Bell Laboratories, demonstrated an important step toward optical computing: a digital optical processor. The optical computer makes use of an optical switching device called SEED (self-electro-optic effect device), developed at Bell Labs in 1986.

Since light beams do not interact, it is hard to make them turn one another on and off. SEED overcomes this difficulty in the following manner: A voltage is applied to the switch which makes it transparent, permitting a laser beam to shine through. Directing a second beam at the switch material makes it opaque.

Currently the prototype is several cubic feet in size and a little bit slower than today's supercomputer. It is anticipated that the next version could be reduced to several cubic inches and made to process as quickly as today's supercomputer. Of course, this optical computer has the potential to get much smaller and to operate at much greater speeds.

Forecast for Optical Computers

General-purpose optical computers are expected to be available commercially early in the next century. Bell Laboratories thinks we will see the first application of an optical computer before we are halfway through the decade of the 1990s.

Of course many obstacles must be overcome before optical computers become practical, and there are plenty of skeptics who believe this will never happen. The skeptics admit that it is possible to build an optical computer, but they do not think it is a practical alternative to the electronic computer. IBM did a lot of research on optical computers in the 1960s and concluded that light uses more energy than electronic computers and would not be practical. Bell Labs representatives think optics are good for transmission and connections, but not for much else.

Before optical computers can become practical, they must be miniaturized so they can be mass-produced. We also must learn how to program parallel processes better before we can take full advantage of the optical computer.

Molecular Computers and Other Advanced Architectures

Of all the advanced, revolutionary technologies, the molecular machine is probably the most exotic, with perhaps the potential for the greatest performance gains, but probably the farthest from reality today. In *Engines of Creation*, K. Eric Drexler[6] discusses the possibility of using molecular computers. He calls it "nanotechnology" because molecules have the dimensions of nanometers (billionth of a meter). One nanometer is a thousand times smaller than the scale of the present semiconductor chips.

Breakthroughs are expected before the end of the century that will enable us to custom-build single molecules that can store and process information and fabricate other molecules. Molecular machines already exist in nature. Examples include DNA, RNA, and enzymes that assist in the reproduction and repair of biological organisms. Scientists have succeeded in developing simple molecular machines in the laboratory by modification of these natural machines. These early, simple man-made molecular machines are only capable of performing very specific tasks, such as artificial antibiotics and interferon. But in the future there will be new types of these systems capable of performing many functions, only some of which may be biological. These new machines will combine the fields of molecular biology, thin-film chemistry, high-temperature superconductivity, flat-screen technology, X-ray lithography, charge-coupled devices, neural network architecture, and atomic scale microscopy.

Types of Molecular Machines

Molecular machines will probably divide into those that process only information (computers) and those that process materials or energy (robots). Molecular computers would replace semiconductor logic devices with molecular logic devices, although we may see an intermediate technology, such as quantum effect devices, along the way. The molecular logic devices would have linear scales a thousand times less than the present semiconductor devices, and would be three-dimensional rather than two-dimensional. Eventually, they will probably be a thousand times faster and have a billion times greater capacity than present microprocessors and RAM. We can envision a parallel processor of the future being the size of a human brain and having more than a quadrillion nodes. The robotic variety could be capable of programmed movement and manipulation of the environment. They could perform functions on other molecules, such as acquisition, storage, transport, fabrication, or repair. They could mass-produce useful molecules from raw chemicals.

A special case is in the area of I/O and communications. One of the functions of active elements on the outside layers of a lattice design might be to change their response to light so that the entire outer surface of the molecular logic device would function as a holographic interface with the outside world. It could therefore be used as a display or as a high-bandwidth channel for I/O. It might also serve as both a camera and a display so that a pair of such molecular logic devices could be connected over a high-bandwidth channel so that each could display what the other sees in full color and three dimensions.

Possible Evolution

There are many different scenarios for how these molecular engines might evolve. For example, extension of current work in molecular biology might lead to highly modified cells, viruses, or viruslike molecular effectors that could lay down at least the substrate for re-creating other, similar devices. This might then be further structured or permanently "nanocoded" using other such modified bioforms. Another possibility might be to use the tendency for certain polymers, such as some lipoproteins, to form highly regular two-dimensional lattices when stretched into molecular films. Perhaps we could structure such a lattice using X-ray scanning or scanning tunneling microscopy.

Conclusion

As computers continue to permeate society, their impact will grow. Like the automobile and television, they will dramatically and fundamentally alter our way of life. Their impact will affect all facets of our lives.

Recreation and entertainment will become more interactive. They will probably even be delivered, viewed, and billed through computers and computer networks.

We already have electronic translators with voice response output for the future traveler, and electronic pocket dictionaries and spellers for the budding author. Next we will have translating telephones and portable expert advisors.

There will probably be an increase in electronic shopping (shopping through television and electronic computer-based catalogs). Marketing and advertising will integrate with these services and become more directed, focused, and customized.

Education will be more individualized, with greater dependence on electronic tutors and interactive instruction via "smart" television.

Work will be less structured, more creative and thoughtful, with more emphasis on group effort.

Computers may include the ability to transport the user into an artificial reality.

The nature of computer software development will become more one of specifying, analyzing, and integrating pieces of packages. To do this kind of activity well will require greater knowledge of the available packages, their strengths and shortcomings, their availability, reliability, and use, as well as where and how to patch and interconnect.

Even the law will be affected. We will have to consider the rights of consumers and the liabilities of vendors when the computer package gives advice that can affect you or your company's health and economic well-being.

Security and protection against electronic intrusion and fraud will become increasingly critical and sophisticated as dependence upon computers in every facet of life grows.

Computer technology may go through some revolutionary changes. In the future we may all be using computers based on superconducting Josephson junctions, optics, or even molecular computers. The computer's architecture may include large arrays of parallel processors all operating in unison.

Don't throw away your old semiconductor textbooks quite yet. Researchers face a difficult nontechnical challenge in displacing the current computer technology: They are fighting a moving target. While they are doing research, the older revenue-producing technology keeps improving. In fact, it was the anticipation of advances in silicon that led IBM to drop the first Josephson junction research. So it is not certain that superconductors and Josephson junction technology, or optics, or molecular computers, will succeed in playing a major role in the computer of the future.

References

[1] *Personal Computer World.* October 1988.
[2] Zachary, G. Pascal. "Artificial Reality: Computer Simulations One Day May Provide Surreal Experiences." *New York Times,* January 23, 1990, pp. A1, A9.
[3] Davis, Stephen G. "The Superconductive Computer in Your Future." *Datamation,* August 15, 1987, pp. 74-78.
[4] Hooper, Laurence, and Schlesinger, Jacob M. "Is Optical Computing the Next Frontier, Or Just a Nutty Idea?" *Wall Street Journal,* January 30, 1990, pp. A1, A2.
[5] Markoff, John. "Laser Device Could Be Key to Faster Computer." *New York Times,* January 30, 1990, pp. A1, D8. (a).
[6] Drexler, K. Eric. *Engines of Creation.* Garden City, NY: Anchor Press/Doubleday, 1986. ■

Cooperative Processing Issues and Trends

In this report:

Datapro Summary

Although it is not commonplace in organizations just yet, cooperative processing is a technology whose time has come. By optimizing a company's computing power and hardware investment, managers are beginning to realize that cooperative processing makes good economical sense. Since the technology is new and cooperative processing tools are unsophisticated and relatively scarce, implementing a cooperative processing environment can be a daunting operation. Still, organizations that move to a cooperative processing architecture find its rewards are plentiful.

Cooperative Processing Benefits

Most of the computing world—anyone running linked mainframes, minis and PCs—is probably spending more time, money and processing power than it needs to and getting less integration and ease of use than it should. Cooperative processing can change all that.

Cooperative processing can cut host resource costs by as much as 60% to 80% and substantially reduce communications costs as well. It can replace crowded, monochrome 3270, AS/400 and VAX screens with a single, colorful, graphical, user-friendly, PC-like interface. That single interface can provide simultaneous, user-transparent access to every essential corporate system while providing for the control, integrity and maintenance of every corporate PC. Cooperative processing is the technology of tomorrow, but companies can start putting it in place today.

This Datapro report is a reprint of "To Each Its Own" by Kathleen Melymuka, pp. 66-75, from *CIO*, Volume 4, Number 7, April 1991.

Cooperative processing means processing different parts of a single application on different computers to make the best use of the processing power of each. It is not a new idea, but it's an idea whose momentum depends upon a critical mass of host-based PCs, which only recently has been achieved. "Cooperative processing has been used with programmable workstations for 10 years or longer, but the dominance of some PC architectures has delayed its broad use," said Allen A. Kalkstein, a partner in the Chicago-Great Lakes management consulting regional office of Ernst & Young, which is currently implementing cooperative processing systems for large corporate clients.

Kalkstein explained that the explosion of stand-alone PCs in the '80s and the ensuing rise of departmental computing put cooperative processing on hold until recently. Today, a large installed base of PCs hooked to hosts, a new understanding of the limits of departmental computing, and IBM's SAA announcements all are pointing to cooperative processing as the future of computing.

A Matter of Economics

Cooperative processing makes the most of an organization's computing power and investment in hardware. "Most people using

Begin with an Interface-Lift . . .

Cooperative processing can be implemented in three phases, starting with the front end, according to Allen A. Kalkstein, a partner at Ernst & Young, a New York-based consultancy that helps large corporate clients develop cooperative-processing systems. "Users want functionality and ease of use, so you start with what we call 3270 beautification. You look at the old boring screens and enhance them with color, pop-up menus and [context-sensitive] help. You take one screen and parse it out to multiple screens; take multiple screens and put them together according to the way work is done."

But true cooperative processing is more than just putting on a pretty face. "In 3270 beautification," said Kalkstein, "you're still dedicated to the mainframe. If it goes down, you go down."

So the second phase, according to Kalkstein, involves "exploit[ing] a full-blown peer-to-peer cooperative-processing environment. You want to delegate as much processing as possible and make it stand alone on the PC." For instance, a marketing application in a company with dispersed sales offices would bring the enhanced mainframe application to the field-office PCs. the field personnel can work on it without any connection to the mainframe, exchanging information with the host only as needed.

Only when the business is completed and it's time for other members of the corporation to act on it do you transmit it and interface with the mainframe applications.

"You want to use the mainframe as a central database and repository," said Kalkstein. "You want to allow extensions of the organization the freedom to act as independently as possible."

The final phase is control. "You've got to maintain and control versions of software and related data sets that run on the PC," Kalkstein explained. His solution is distribution software that transmits new applications, releases, tables and documents to be sure everyone is in sync. "Without those three pieces, cooperative-processing solutions will only be partially useful to an organization."

---

host-based applications on the PC are using a PC as a dumb terminal," said Phllip Carpenter, product manager for Mozart Systems Corp. of Burlingame, Calif., a maker of cooperative processing tools. "That is not a strategic way to take advantage of the PC. They are powerful little beasts."

"The biggest reason for cooperative processing is economics," said Douglas Engle, product manager of Vision 2000, a collection of cooperative-processing products designed to enhance management productivity for Software 2000, based in Hyannis, Mass. "The mainframe is very expensive from a human-resources standpoint. It can take 30 people just to operate a mainframe. The PC is low-budget equipment, and the people are already there. Everything you can pull off [the host] and put on little boxes gives a higher return on your investment."

With mainframe MIPS running around $150,000 each and PC MIPS below $1,000, the economics are clear, said Stephen Sayre, vice president of marketing for Easel Corp., a Woburn, Mass., maker of cooperative-processing tools. "It's an order of magnitude differential," he said. "Of course, a CIO will tell you that a MIPS is not a MIPS is not a MIPS, but when the smoke clears, if you're off-loading from mainframes to PCs you're going to save hundreds of thousands—if not millions—of dollars over your investment horizon."

Cooperative processing combines the best of the mainframe and PC worlds. The PC can provide colorful, user-friendly graphic interfaces, editing, data validation and a good deal of local processing. The host is reserved for big-guns number crunching and integrating PC transactions into larger corporate systems. Because as much processing as possible is done at the PC, expensive mainframe cycles are replaced by cheaper PC cycles, and less interchange of data means lower communications costs. "The more you can offload to the PC, the more money you're saving," said Mozart Systems' Carpenter.

Cooperative processing also reduces training costs, since the single, common user interface provides the same look and feel to each application, allowing simultaneous, user-transparent access to a range of mainframe or mini hosts from the PC. Using a tool such as Mozart, for example, you can pull data from 3270s, AS/400s and DEC VAXes simultaneously. "The user has no idea this is even going on," said Carpenter. "Before, you would have to know where the data is, log onto each different machine, write down the data and log off. Now you don't have to know or care where the information is."

Cooperative Tools

The logical place to start cooperating is with the user interface. "You don't want a $5 million box doing something a PC is designed to do," said Engle. "Machine cycles get soaked up in the display process. You're sending machine cycles back and forth saying, Now the display looks like this. Now the display looks like this. . .

Cooperative processing tools such as Infront by Multi Soft Inc. (Lawrenceville, N.J.), Mozart by Mozart Systems Corp., and Easel by Easel Corp., enable developers to construct user-friendly front ends to mainframe programs. Since the front ends intercept the terminal data stream sent by the host, they look to the host like a dumb terminal, and no changes to mainframe code are required. Such tools commonly include a third- or fourth-generation language, a screen-development tool, a communications tool and a software-distribution tool. These can assist developers in shifting processing tasks from the mainframe to the PC, but that's the tricky part. "It's a fundamental rethink of how you build applications," said Easel's Sayre. "Instead of building skyscrapers you're building one-story modular structures on a million square feet of land. You've got to figure out which part of the application belongs where."

Design isn't the only problem. The immaturity of cooperative processing presents its own challenges. "It's a new technology, so there will be some of the same problems you get with new software," Kalkstein explained. "Releases may not be fully ready. It's like the late '60s and early '70s [when current mainframe technology was in its infancy], not like the maturity of [today's] mainframe environment. In a way we're slipping back to a period where vendors are less sophisticated."

Anatomy of an Implementation

At Comdisco Inc., marketing drives business, and a jerry-rigged system of PCs linked to the mainframe was driving the marketing department crazy. The Rosemont, Ill., company buys, sells and leases high-tech equipment, so when the inefficient system threatened to swamp sales in 1987, Ron Markham, hired to build a central corporate database, became the cooperative-processing guru by default.

The sales-support system was complex and difficult to learn—a mishmash of automated and manual processes, many of which were redundant. For example, a purchase leaseback transaction could involve any of 300 kinds of paperwork depending on terms. A user might have to enter the same information as many as 23 times for one transaction.

Markham realized that cooperative processing might provide a solution to this paperwork morass. "It was a classic example of a business need driving the technology rather than the other way around," he said.

Markham chose Infront, a cooperative-processing tool from Multi Soft Inc., of Lawrenceville, N.J. "They were essentially selling a 3270 beautification product . . . [in which] you take a CICS application, run Infront on PC emulation, capture the mainframe screen and map it to a user-friendly screen on the PC, so you'd get pop-down menus, color, graphics, editing. That's not true cooperative processing, but we built the first marketing application using that technology."

Markham built a classic mainframe application and put Infront on the PC as the user interface. "It was totally dependent on the mainframe," he explained, "but we got a feel for the product, marketing got a feel for the types of screens, and we started working toward a real cooperative-processing application."

The intended users demanded that the new system reduce the training curve drastically. They wanted no user manual. Although it would be necessary to use information from the corporate database, they wanted to be able to do work if the mainframe or the network went down. They wanted to continue to use WordPerfect, their word processor of choice. They wanted high productivity.

The team built in context-sensitive tutorials to guide users and a tree structure that minimizes decision-making by automatically bringing up the proper forms triggered by the information entered by the users.

Customer information stored on the mainframe is now also stored on the PCs of the appropriate salespeople, so users needn't access the host. If a support person logs on and the mainframe is down, he or she can still generate a contract. The PC stores the appropriate information until the mainframe goes up, then passes it on for integration with other corporate systems.

One of the most difficult pieces was the design of a seamless interface between Infront and WordPerfect. "That was tough," said Markham, but with help from Multi Soft, WordPerfect and the consulting team, it finally came together.

With 200 PCs at 23 offices throughout the U.S. and Canada, Markham knew that control of the system was crucial. A software-distribution facility runs on the host and the PCs, controlled by a master PC at corporate headquarters. "Anything new goes to the master PC, and it's uploaded to the mainframe for distribution to the [rest of the] PCs," said Markham. "When a user logs on, the PC automatically goes to the host to see if there's anything new. It's all transparent to the user."

All of this didn't come easily. "We used a lot of outside help to get people up to speed on the technology," said Markham. "[Consultants from] Ernst & Young put together the basic infrastructure. It's a manpower-intensive technology."

Being a pioneer in a technology brings its own set of problems. "The Infront product wasn't all that mature, but it was still the best thing out there," Markham recalled. "They were thin on quality assurance so we had to watch for land mines. We developed a very structured programming environment with a lot of rules and quality-assurance checks. We also kept a high-level view of the product we were trying to build, so if we were wrong, at least we were consistent and we could change it later across the board."

Multi Soft recently released a new version that addresses many of the issues with which Markham's group struggled. "We tested a lot of code for them," he chuckled. When the first system was ready to run, Markham faced another hurdle. "We had problems with performance because of all the layers of software we had put together," he said. "The XTs were just too slow. We had to upgrade them with '386 processors to make it a useful product."

Markham's group has completed three major products and expects to complete a new one each month during the coming year. Reaction to the products has been intensely positive. "Usually you expect resistance to a new system," said Markham. "The only complaint we get is that it's frustrating for them to wait for the rest."

The Wave of the Future

Since the technology is new and the products are still somewhat immature, most implementations depend on outside help. "It's not as simple as vendors would like to portray it," admitted Sayre. "It's a new architecture. Developers will have to take a lot of baby steps in the beginning. Distributing the interface is a good baby step to start with, [but] the typical sale involves consulting."

Despite the problems, experts agree that the benefits of cooperative processing are worth it. Ultimately, they say, cooperative processing will change both computing and business. First, it will erase the line between personal computing and enterprise computing in organizations. "Personal computing won't be personal anymore," said Sayre. "Personal computing will be a part of enterprise computing. The days of islands of information are being eclipsed."

More important, cooperative processing is the vehicle for bringing computing to the source of the transaction, blurring the line between those inside the organization and those outside. "Many companies are trying to put data entry at the source so the customer can create and access his own data whenever he wants," said Gerry Jurrens, vice president of Multi Soft. "[Cooperative processing] cuts out the middleman."

With its dispersed computing and central control, cooperative processing will enable more and more organizations to move toward this model. "Cooperative processing is here, and it's the wave of the future," said Peter Keen, executive director of the International Center for Information Technology in Washington. "First you need cooperative business processes. Cooperative processing is the architecture to support that. This is going to be the most important issue in the '90s." ■

Why Client/Server Computing?

In this report:

Datapro Summary

Client/server is a computing model that divides a network's functions into front-end and back-end functions. Front-end functions consist of handling the user interface and presentation of information to users; back-end functions are concerned with fulfilling requests for information, sorting, and performing computation on the data requested. This approach reduces the errors and breaches in security that can occur with computing models that require downloading all of the files necessary for a request, then uploading the files after a user has manipulated and changed them. There are a number of business benefits of client/server computing, such as distributing computing across the network, providing transparent remote access to computing resources, offering lower-cost computing platforms, and increasing communication among users. The report offers several guidelines for choosing server hardware and software, including how to evaluate a system's performance, reliability, and availability. Client/server computing offers management better control over security, more flexibility to migrate to other software and hardware, and a more hierarchical network framework that is easier to support.

Frank is the network manager of a major Chicago bank. The consumer and retail banking departments favor high-speed token ring networks. Each ring has its own print server. In addition, the entire regional banking division has five electronic mail servers, two communications servers (X.25 links, primarily), and six file servers. The international banking division favors Ethernet, with more file servers and fewer mail servers. They also have three multipurpose servers that provide both file and communications service.

Valerie manages the network of an electronics manufacturer in Atlanta, Georgia. For their four-building headquarters campus alone, she needs a staff of twenty network specialists that spend most of their time managing the cable plant and the company's servers. Each building has at least twelve file servers, seven mail servers, and three communications servers. In addition, each building has at least two multipurpose servers that combine file and mail service in a single box. Valerie's responsibilities include managing the interface between this headquarters campus and the forty national and nine international sales/support offices. She has a multipurpose server at each of the U.S. regional offices.

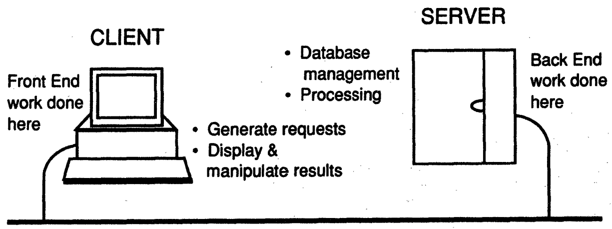
Do you think Frank and Valerie's complex, server-intensive internets are using client/server computing architectures?

They aren't.

> "It's not so much the things we don't know that get us into trouble; it's the things we know that ain't so."
>
> —Mark Twain

This Datapro report is a reprint of "Why Client/ Server Computing" pp. 41-64, from *Client/ Server Computing,* Davis Fields, III, editor.

Figure 1.
The Client/Server Computing Architecture



Client/server computing is one of the least understood of the much-talked-about networking topics today.

Let's start by understanding what client/server computing is not. It is not:

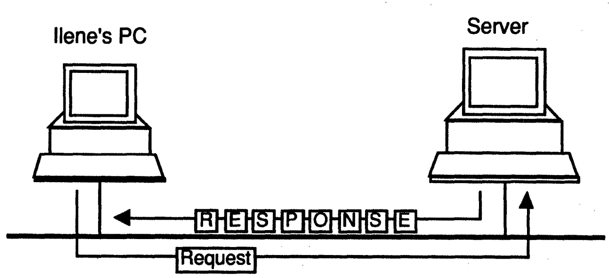
- Any network with a server on it
- Any network with a lot of servers on it
- A more advanced kind of server

Client/server is a model of computing that divides functions between client and server machines. The server focuses on tasks that are compute-, disk-, or memory-intensive and often involve management of large amounts of data. The clients focus on user interface and decision-support functions, sending requests to the server and displaying or processing the results. The server is sometimes described as running the "back-end" functions, and the clients the "front-end" ones. The separation of functions into front-end and back-end processes is shown in Figure 1.

Each part—both server back-end and client front-end—can be developed, migrated and serviced separately. The parts are developed so as to optimize a particular function, either database retrieval or information-display so each device in the network can be utilized most effectively. Both the use of computing resources and the use of corporate funds are leveraged with low-cost, high-performance server platforms and a preserved investment in hardware, software and training.

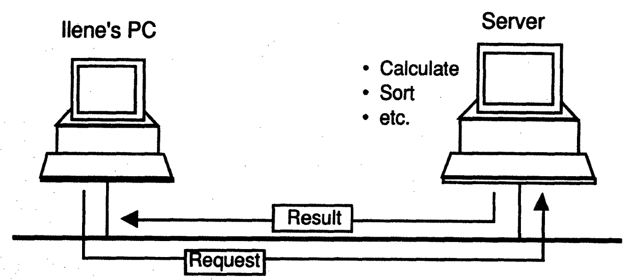
Let's look at a typical data management transaction: Ilene in Product Marketing wants to sort through pricing data to find the optimum mix of base price and add-on prices for her new products. She also wants to investigate sales reports, client lists and billables history. In a traditional server-based network, Ilene will have to transfer from the server all the files in which that information is located. This results in large amounts of network traffic, as

Figure 2.
File Server Architecture



Ilene's request generates a lengthy response in a file server architecture.

Figure 3.
Client/Server Computing Reduces Network Traffic by Transporting Only Requests and Results



shown in Figure 2, as well as raising security and data integrity concerns. Ilene will then have to do the sorting and calculating on her own desktop computer, which is typically not as fast as larger machines on the network. If she makes changes in these files, the files must traverse the network again, raising additional traffic, security, and data integrity issues. This process makes very inefficient use of the network, the PC, and the server.

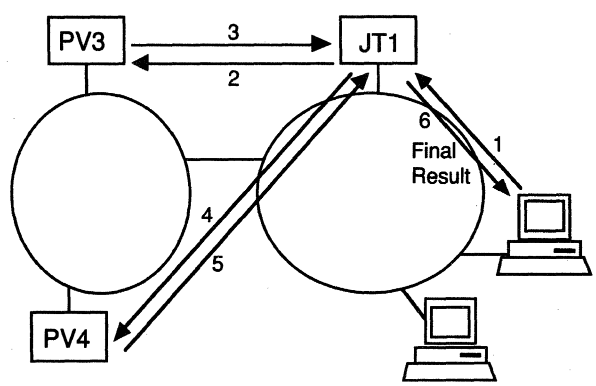
In the client/server computing scenario, only requests and results are transmitted (see Figure 3). In our example, the server receives Ilene's request and performs searches for each piece of information she needs. Only the results of those searches are sent back across the network to Ilene. This results in a much more balanced and efficient use of the network, the PC, and the server.

Client/server computing allows the client to ask the questions and lets the network and servers locate and deliver the answers. A server can even transparently contact other servers or systems to obtain information for users. In Figure 4, network server JT1 gets a request from an end user and retrieves data physically scattered across the network. JT1 transmits its needs to other servers, which do their own sorting and calculating before transmitting results to JT1.

The client/server computing model assures end users of transparent data access across the entire network, regardless of what hardware or software platform the information may be stored on. In a client/server network, the entire network's computing power is available to each user, to be called upon when needed.

The file, mail, print and communications servers most LAN users are familiar with do not represent client/server

Figure 4.
Server JT1 Searches Servers in Other Physical Locations in Response to a User Request



architecture because they do not have this division of function built into their hardware or software. While they provide useful and convenient features for small groups of PC users, they do not address the needs of most large organizations to integrate their PC networks, larger systems, and mainframes into useful information management tools.

Putting back-end servers in charge of data management and computation and leaving front-end clients free to concentrate on user interface and presentation needs has tremendous advantages for network managers and their end users. Let's take a look at some of these business advantages of client/server computing.

Business Benefits of Client/Server Computing

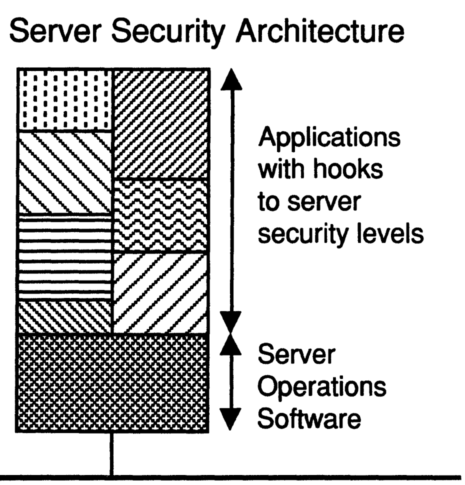
- Leverage existing resources
- Distribute computing across the network
- Transparent remote access to computing resources
- Lower-cost computing platforms
- Increased communication among users
- Lower network management costs
- Lower network traffic volume
- Increased security
- Centralized data management
- Increased flexibility

Let's look at some of these business benefits in more detail.

- *Effective distribution of computing power.* Effective utilization of computing power, wherever it is in the network, means that one machine isn't overloaded while another sits idle. This delivers obvious cost benefits.

A major business advantage of client/server computing derives from the separation of functions into front-end user devices and back-end server devices. In a client/server network, the processing and decision-making power of every device in the network is being most efficiently utilized. With a typical file server, the slowest processor in the entire network (the user's PC) is doing the bulk of the work, while the user waits. With user devices that can concentrate on their user interface and their desktop decision-making processing and servers focusing on data storage and retrieval management, your business' use of computing resources is leveraged.

Figure 5.
Server Security Architecture Lowers the Burden on Individual Applications

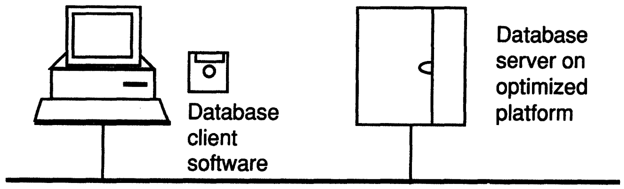


How Critical?

"Mission-critical" is another one of the overused and wrongly used phrases in many business discussions. Every business has a different sense of what is critical to maintaining daily revenue flow and operations. Stock exchanges, air traffic control networks, and utility networks all have fairly obvious mission-critical applications. For a commodities or currency broker, the buy/sell application would be in this category. Mission-critical applications are sometimes defined as those which must run continuously, or will cause great damage to the company if interrupted even briefly. A larger class of applications can be categorized as "business-critical," meaning they are essential to the business but that short interruptions in service can be tolerated. A business-critical application needs high availability and high performance, at a reasonable price. These applications reside in the middle ground between the nice-but-not-necessary and the can't-fail applications and correspond to most companies' business needs.

- *Lower cost computing platforms.* Client/server computing allows mainframes to function as enterprise data management and storage systems, while it moves much of the daily activity to lower-cost server platforms. Server performance, based on 1990s hardware and software technology, is considerably less expensive than equivalent mainframe performance.
- *Increased communication among users.* A LAN equipped with file servers still represents mini- or mainframe-based computing because these servers are relegated to auxiliary functions. Without the full force of distributed computing power, users still consult mainframes and minicomputers for data management; cooperative computing (with groupware as an example) is impossible.
- *Better utilization of network bandwidth.* The proliferation of high-performance user devices means that network traffic is increasing; as an example, sending a color image the size of a computer screen can require eight megabytes of data to traverse the network. Because client/server computing reduces the amount of data moved around, it can bring important advantages to large or crowded networks.
- *Separate development saves money and gives you flexibility.* The front-end and back-end components of a client/server network require different skills to create and maintain. A software designer who is an expert at

Figure 6.
The Use of an Optimized Server Platform for a Client/Server Database



friendly user interfaces isn't necessarily an efficient designer of a relational database management system (RDBMS), for example. If vendors can separate these two very diverse functions in the development process, you will end up with more specialized and better-designed software. The separation of the development work means that these two components can also be serviced, maintained, upgraded and migrated transparently to one another.

- *Lower network management costs.* Because of their modular function in the network, servers are more easily migrated, expanded, serviced and maintained than mainframes, significantly lowering your maintenance costs. The money you invest in security is also well leveraged in mainframe-type protection without proprietary, expensive mainframe solutions.

Finally, it's easier to customize performance of a server than of a typical mainframe. Being a simpler, more maintainable platform gives a server a considerable advantage in customization. And of course you always want to be able to customize your network tools to the specific circumstances of your company's business.

- *Enhanced security, as well as more flexible security options.* Security is enhanced in client/server networks by letting the server enforce security rules. Each application's security requirements can by programmed into the server's operations software, as is shown in Figure 5. Then the applications designers can concentrate on maximizing features and performance—solving the users' problems—and let security be a server function. The applications designers do not have to incorporate security into the actual application code so it can be that much simpler, faster and error-free.

Figure 7.
Groupware in Action

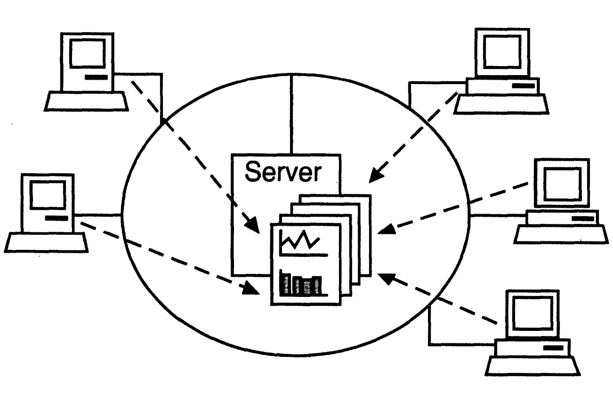
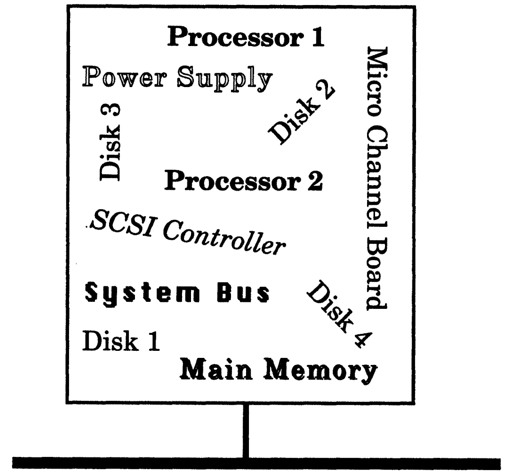


Figure 8.
A Server's Components



Choosing a Server

If you are embarking on a project to implement a client/server database, you need to carefully choose a platform on which to run the database server software. Although a PC-based system might be able to perform this function, most industry experts are recommending the use of systems designed exclusively as servers.

A system in which both hardware and software are optimized for server operations should provide better features and performance than a platform originally designed as a front-end or client machine (see Figure 6).

Choosing Software for Your Client/Server Network: Groupware

Lotus Notes is one of the first examples of a new category called groupware, designed to facilitate communication and electronic collaboration among members of workgroups. Groupware makes it extremely easy to attach documents to each other and add numbers, graphics and other imports to documents, simulations and presentations. In Figure 7, we can see the New Business Development group of an insurance company brainstorming for an upcoming presentation. The contributors are adding graphics, voice annotation, simulations and text to a series of documents that will then be distilled into recommendations.

The exciting aspect of groupware is that it allows groups of people to get together to work more productively. A desktop computer only helps one individual work better and faster. With the rise of client/server computing, we now have software that gives an entire workgroup or department a chance to work cooperatively to achieve business results. The real glory of using computers in business is having users (through their computers) communicating with other users to increase group productivity and deliver better goods and services to customers. Until now, we have only been inching our way toward this goal. With the dawn of groupware, we can actually start realizing the value of the thoughts, insights and tangible work results of every member of a team.

Server Performance

The server's ability to respond quickly and reliably depends on the quality of the processing engine it contains.

Multiprocessing architectures are beginning to be employed in server architectures to meet the growing performance demands of users.

Performance Criteria

The point of all performance measures is really to measure how fast the user will see results. Unfortunately, the measures of performance that are usually applied to servers include the maximum number of transactions per second. This may not be an accurate assessment of how well the server will perform for you in your network, for a number of reasons:

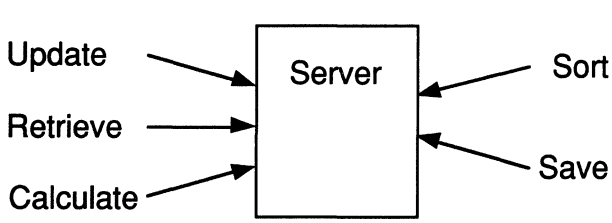
- *You need to know how fast the entire system is, not just how fast the fastest component is.* Figure 8 shows a hypothetical network server system and its various elements. The fastest element is probably the processor, but a fast processor can only deliver performance as fast as the I/O channels and the buses can move the data. In other words, a system moves only as fast as the slowest element in that system. If you want to know how fast the system really is, clock the slowest element.

Balance is the key to useful performance in the real world. In order to have a truly powerful system, you need balance between computing power, disk memory, input/output (I/O) and network throughput.

- *You need to know how fast the server system works under your typical load levels, with your particular mix of traffic types.* You need to examine how the server system operates in an environment as close to a working network as possible. That means lots of people, doing lots of different things, as shown in Figure 9. You know what your users are currently doing in terms of traffic mix. Insist on a benchmark with your traffic mix, or something close to it. A processor that can do one thing very fast may have trouble switching gears (due to slow internal buses, for example). It may also have trouble performing multiple types of tasks concurrently, for instance printing files and serving applications.

You need to look at benchmark tests that simulate the number of users and the mix of device types in your network. If you are a network manager for a major New York bank, for example, you have over 10,000 personal computers and several hundred multitasking workstations attached to your network. What will happen to this server if even a small percentage of your actual number of users tries to log on? The system you are considering buying may be the fastest processor in the world with only seven users logged on. Unless your network really only has seven users—and all PC XTs, at that—you really don't know how it will react.

Figure 9.
A Typical Network Traffic Mix for a Realistic Benchmark



SQL

To allow compatibility between database software products, developers needed a common language, called a query language. IBM's Structured Query Language (SQL) has been adopted by most software vendors as their standard method of communication when database packages need to exchange information. Novell, Microsoft, Oracle, Sybase, Informix, Gupta and Borland International have all adopted SQL as their "translator."

- *You need a server system that will degrade slowly and gracefully.* If you can't get a benchmark that approximates your real network conditions, at least inspect the published benchmark closely. Many benchmarks in use show a peak in performance when the number of users is small, often less than ten. The more important data for you is how the system will perform in your environment with your number of users, which is typically much larger. A carefully designed, balanced system will continue to exhibit high performance even with large numbers of users. Figure 10 shows performance curves for three servers with similar "peak" results but very different performance at fifty users and above.

Reliability and Availability

Some business managers talk about reliability, availability and fault tolerance as though they were synonymous. They aren't. Reliability refers to how often a particular hardware component fails and the usual measure is mean time between failure (MTBF). In order to have a reliable system, all the components must be reliable. Thanks to advances in modern hardware design and manufacture, most

Figure 10.
Looking at the Entire Performance Curve

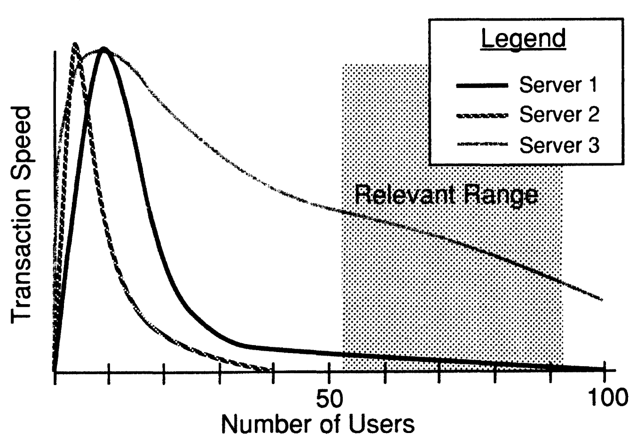
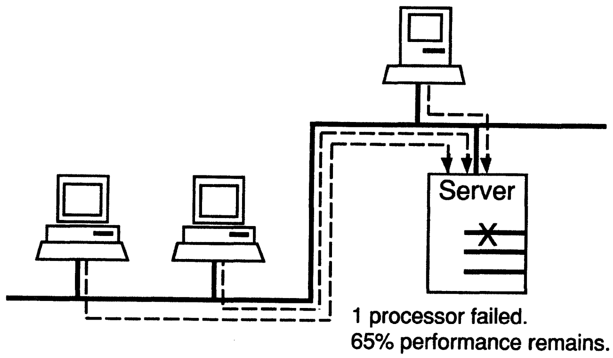


Figure 11.
A High-Availability Server Continues to Work After Processor Failure



hardware components are very reliable, so most systems can claim a very high hardware reliability.

High Availability Systems

Availability measures the percentage of time the system is available to users. High availability means that when the system fails it has self-healing mechanisms in place that allow it to return to service very quickly without human intervention. In addition to self-healing attributes, a system could be rebootable or downloadable from a remote site, or it could have tools which notify administrators of problems, so they can be corrected before they become serious. A high availability system could be designed to continue functioning in a reduced configuration after a failure (see Figure 11).

In this scenario, the server could keep functioning in the face of a serious hardware failure, but more slowly (as it works around the failed component using redundant pathways). This allows the server to continue to be available to users while the parts are swapped. Ease of repair is part of the high availability equation. The servers that can continue working in a degraded condition avoid the all-or-nothing pitfall common to PC-based servers.

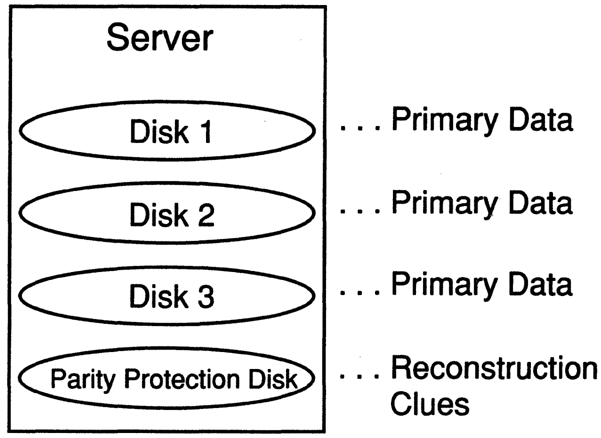
A high availability system doesn't claim to never, ever fail. It does claim to come back up into service very quickly. A highly available system would be appropriate for both mission-critical and business-critical applications and commands a modest price premium for that feature.

A fault-tolerant or nonstop system pushes the availability envelope to its extreme by making every component redundant, for a correspondingly steep price tag. This level of insulation from failure is mandatory for stock exchanges, utility networks and commercial installations where the cost of downtime is measured in nonrecoverable millions or billions of dollars. (Infonetics of Santa Clara, CA, estimates the cost of computer downtime for a *Fortune* 1000 company at over $4 million/year.) It is appropriate but not mandatory for other kinds of banking, stock and commodities transactions but the heavy price tag makes it inappropriate for most ordinary commercial enterprises.

Implementing High Availability Systems

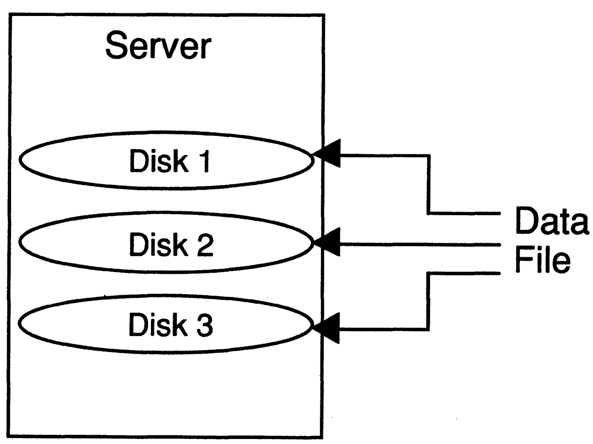
Technologists have come up with several ways of bringing high availability to the point where it can be implemented

Figure 13.
Parity Disk



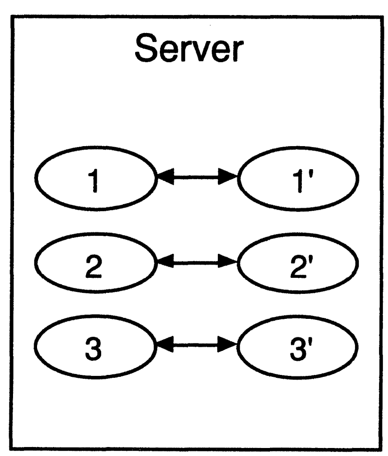
The parity disk allows your server to reconstruct a lost disk's data.

Figure 12.
Striping



Striping speeds access time by allowing simultaneous read/write on a file.

Figure 14.
Disk Mirroring Replicates Every Disk with Byte-for-Byte Backup



in working networks. These new developments cluster around component redundancy, data integrity and serviceability.

Redundancy of components is a significant issue in server availability, especially redundancy in CPUs, controllers and network interface boards. Components must also be easy to service if you are to have a highly available system. Error notification and remote servicing improve availability due to rapid response.

When we move to reliability of the data on a disk, techniques have been developed to reduce the risk of data loss. While a user continues to "see" one hard disk drive on a server, software can actually stripe or break up the data into chunks (see Figure 12) that will be physically placed on multiple disk drives. The goal here is to use multiple disks, together with a parity protection algorithm, to allow the system to continue operations after a disk failure. In the event of a failure, your software can re-create what was on the failed disk by looking at the remaining data pieces that are scattered (via striping) throughout the other disk volumes (see Figure 13). The server's performance will degrade while the missing disk is reconstructed, of course, but the system is self-healing and the process should be relatively transparent to the end users.

RAID (redundant arrays of inexpensive disks) is a specification of the redundancy and reliability of multiple-disk subsystems. The five levels of RAID start at level one with total duplication of every disk, a process called disk mirroring (see Figure 14). Most popular systems achieve level three function. The ultimate goal, or level five, decreases the penalty for reliability to a minimum while increasing performance.

Management in the Client/Server Network

The client/server computing model helps bring essentially flat networks into a more hierarchical framework for easier and more effective management. As networks grow in size, segmenting managed resources into logical management domains allows managers to automate some tasks, zero in on problem areas faster and spend time analyzing and planning rather than just fire fighting. This makes client/server networks easier to support.

Security, Migration, and Flexibility

Client/server computing increases security significantly by reducing the movement of sensitive information across the network and by enforcing security rules at the server. We have already shown how this is preferable to traditional computing architectures, where security must be addressed by individual applications.

Client/server computing also allows you to migrate to new technology gracefully. Monolithic mainframe-based systems are typically difficult to adapt as your needs change or grow, or as new technology becomes available. The client/server architecture, on the other hand, is based on a segmentation of roles between servers, networks, and clients, and allows graceful changes in each without necessitating major redesign. The network can be enhanced with a faster medium, the client operating system can be upgraded, or a server can be replaced with a more powerful model—all independently and all without major interruptions in service.

Optimum Server Design Requirements

- Dedicated to server operations
- Multiprocessing architecture
- High availability features
- Supportability tools
- High performance under heavy load conditions
- Extensibility
- Manageability and security features
- Based on standards wherever possible

The TP1 Benchmark

The most popular transaction processing performance measure, the TP1 benchmark, derives from the Debit/Credit benchmark published in 1985 in a Tandem Technical Report (TR85.2). The TP1 (transaction processing) benchmark measures a server's performance in transactions per second (TPS). The benchmark specifies hundreds of short update transactions, one right after the other. The more "real world" measure of many complex transactions is not covered in this benchmark. The design assumes a mainframe with attached terminals is the dominant network design.

The TP1 benchmark does *not* specify the transport and network protocols used to interact with the server. Some tests use TCP/IP; others use proprietary communications protocols. To compare these benchmarks you must assume that all protocols are equally efficient in carrying transaction traffic, something any network manager knows to be untrue. Consult *Readings in Database Systems* by Michael Stonebraker (Kaufman, 1988) for an in-depth discussion of the technical issues.

One of the critical requirements of a major corporate internet is flexibility. The network manager needs to focus on user needs and let technology track needs. With your client/server network, you will be able to deliver the service that users need, while retaining the control that you as the implementer and manager need. The ultimate goal of the network manager is to achieve effective balance across the network, preventing server performance, network bandwidth, internet devices or user computing platforms from becoming significant bottlenecks. ■

64-Bit Microprocessors

In this report:

Datapro Summary

Although the Mips R4000 is presently the only 64-bit microprocessor, there are more such chips on the horizon. Industry experts predict that the pervasive use of 64-bit microprocessors will arrive by 1995 or sooner. That means faster, large-integer math capabilities, and more expansive virtual memory addressing for servers and desktop computers.

Today's most popular computers are built around 32-bit microprocessors. The next generation of chips—64-bit microprocessors—will bring even more power to the desktop.

But what does it mean to call a chip *64-bit*? It's easy to get confused, because different numbers of bits are used in different parts of a microprocessor (see the sidebar "What's in a Chip?"). Although the Mips R4000 is currently the only 64-bit microprocessor, 64 bits is almost certainly a coming trend. At microprocessor conferences, sessions on the future of chip technology routinely predict widespread use of true 64-bit microprocessors by 1995 or earlier.

You may be thinking, "My PC software still runs in 16-bit mode, and it will be years before the software catches up with 32 bits. But 64 bits? People who predict widespread use of true 64-bit microprocessors by 1995 must be raving lunatics!"

There are two reasons for the prediction: 64-bit integer processing and convenient use of more than 32 bits of address space.

This Datapro report is a reprint of "64-Bit Computing" by John R. Mashey, pp. 135-142, from *BYTE*, Volume 16, Number 9, September 1991. Copyright © 1991 by McGraw-Hill, Inc. Reprinted with permission.

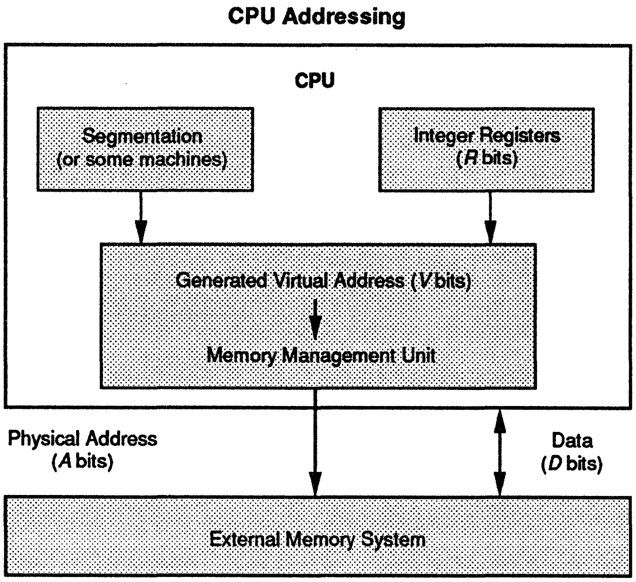
The first reason is a straightforward performance issue; the second has more widespread implications. As you'll see, application for 64-bit microprocessors exist for both servers and desktops.

CPU Architectures

When it comes to CPU architectures, it helps to distinguish between Instruction Set Architecture, which presents an assembly language programmer's view of a processor, and hardware implementations of that ISA. Successful ISAs persist unchanged or evolve in an upward-compatible direction for years. Distinct implementations are often built to yield different cost/performance points. At times people get confused about the difference between ISA and implementation sizes. Table 1 may help clear up the confusion.

In Figure 1, the CPU's integer registers are R bits wide. Address arithmetic starts with R bits, either producing a virtual address size of V bits (V is the generated user address, $V \leq R$) or using a segment register to expand R bits to V bits. The memory management unit translates V bits of virtual address to A bits of physical address that are actually used to access memory. For each access, up to D bits are transferred (i.e., the data bus is D bits wide). For user-level programs, R and V are programmer-visible properties of the ISA; A and D are

Figure 1.
CPU Addressing



usually less-visible implementation-specific characteristics. (Floating-point register size is almost always 64 or 80, and so is not included.)

Table 1 lists numbers for well-known computer families. For simplicity, *V* is given only for user-level programs. The table shows that physical address size (*A*) and data bus size (*D*) can vary within a processor family. The IBM S/360 family included five data bus sizes (8 to 128 bits); the 32-bit Intel 386 is sold in two sizes—32 and 16.

Better Performance With Bigger Integers

For years, PDP-ll UNIX systems have used 16-bit integers for most applications, as do many PCs. Sometimes performance can improve merely by switching to larger integers. Integer code has proved resistant to recent speed up techniques that have greatly helped floating-point performance, so any integer improvement is welcome. Some applications for 64-bit integers are the following:

- *Long Strings of Bits and Bytes.* By using 64-bit instead of 32-bit integers, some programs may run up to twice as fast. First, operating systems often spend 10% to 20% of their time zeroing memory or copying blocks of memory; often, doubling the integer size can help these operations. Second, modern global-optimizing compilers spend a great deal of time performing logical operations on long bit vectors, where 64-bit integers nearly double the speed. Third, the increasing disparity between CPU and I/O device speed is increasing the use of compression/decompression methods, some of which rely on the main CPU, where 64 bits may be helpful.
- *Graphics.* Graphiç applications are a special, but important, case of the long bit-and-byte-string problem. Using 64-bit integer operation can speed the work required by raster graphics. The increase in performance is especially true for large-area operations like scrolling and area-fill, where performance may approach a full two times that of a 32-bit CPU. This approach helps raise the graphics performance of a minimal-cost design—a CPU plus a frame buffer but without graphics-support chips.
- *Integer Arithmetic.* Most chips make addition and subtraction of multiprecision integers (i.e., 64-bit, 96-bit, 128-bit, etc.) reasonably fast, but multiplication and division are often quite slow. Cryptography is a heavy user of multiple-precision multiplies and divides. Financial calculations could use integer arithmetic; 32-bit integers are far too small, but 64-bit integers are easily big enough to represent objects like the U.S. national debt or Microsoft's annual revenue to the penny.

Big-Time Addressing

Perhaps more important than using 64-bit integers for performance is the extension of memory addressing above 32 bits, enabling applications that are otherwise difficult to program. It is especially important to distinguish between virtual addressing and physical addressing.

The virtual addressing scheme often can exceed the limits of possible physical addresses. A 64-bit address can handle literally a mountain of memory: Assuming that 1 megabyte of RAM requires 1 cubic inch of space (using 4-megabit DRAM chips), 2^{64} bytes would require a square mile of DRAM piled more than 300 feet high! For now, no one expects to address this much DRAM, even with next-generation 16-Mb DRAM chips, but increasing physical memory addressing slightly beyond 32 bits is definitely a goal. With 16-Mb DRAM chips, 2^{33} bytes fits into just over 1 cubic foot (not including cooling)—feasible for deskside systems.

An even more important goal is the increase of virtual addresses substantially beyond 32 bits, so you can "waste" it to make programming easier—or even just possible. Although this goal is somewhat independent of the physical memory goal, the two are related.

Database systems often spread a single file across several disks. Current SCSI disks hold up to 2 gigabytes (i.e., they use 31-bit addresses). Calculating file locations as virtual memory addresses requires integer arithmetic. Operating systems are accustomed to working around such problems, but it becomes unpleasant to make workarounds; rather than making things work well, programmers are struggling just to make something work.

The physical address limit is an implementation choice that is often easier to change than the virtual address limit. For most computers, virtual memory limits often exceed physical limits, because the simplest, cheapest way to solve many performance problems is to add physical memory. If the virtual limit is much smaller than the physical limit, adding memory doesn't help, because software cannot take advantage of it. Of course, some processors use segmentation schemes to extend the natural size of the integer registers until they are equal to or greater than the physical address limit.

The Mainframe, Minicomputer, and Microprocessor

Reflect on this aphorism: *Every design mistake gets made a least three times: once by mainframe people, once by minicomputer people, and then at least once by microprocessor* *people.* An illustrative sequence is found among IBM mainframes, DEC superminicomputers, and various microprocessors.

IBM S/360 mainframes used 32-bit integers and pointers but computed addresses only to 24 bits, thus limiting virtual (and physical) memory to 16 MB.[1] This seemed reasonable at the time, as systems used core memory, not DRAM chips. A "large" mainframe (such as a 360/75) provided at most 1 MB of memory, although truly huge mainframes (360/91) might offer 6 MB. In addition, most S/360s did not support virtual memory, so user programs generated physical addresses directly. There was little need to consider addresses larger than the physical address size. Although it was unfortunate that only 16-MB was addressable, it was even worse to ignore the high-order 8 bits rather than trap on non-zero bits. Assembly language programmers "cleverly" took advantage of this quirk to pack 8 bits of flags with a 24-bit address pointer.

As memory became cheaper, the "adequate" 16-MB limit clearly became inadequate, especially as virtual addressing S/370s made it possible to run programs larger than physical memory. By 1983, 370-XA microprocessors added a 31-bit addressing mode for user progams but were required to retain a 24-bit mode for upward compatibility. Much softwsare had to be rewritten to work in the 31-bit mode. I admist that I was one of those "clever" programmers and was somewhat surprised to that a large program I wrote in 970 is still running on many mainframes—in 24-bit compatibility mode, because it won't run any other way. "The evil that men do lives after them, the good is oft interred with their bones."

By the mid-1980, 31-bit addressing was also viewed as insufficient for certain applications, especially databases. ESA/370 was designed with a form of segmentation to allow code to access multiple 2-gigabyte regions of memory, although it took tricky programming to do so.

In the minicomputer phase of this error, the DEC PDP-11 was a 16-bit minicomputer. Unfortunately, a single task addressed only 64 kilobytes of data and perhaps 64 KB of instructions. Gordon Bell and Craig Mudge wrote, "The biggest and most common mistake that can be made in computer design is that of not providing enough address bits for memory addressing and management. The PDP-11 followed this hallowed tradition of skimping on address bits, but it was saved on the principle that a good design can evolve through at least one major change. For the PDP-11, the limited address space problem was solved for the short run, but not with enough finesse to support a large family of minicomputers. This was indeed a costly oversight."[2]

Some PDP-11/70 database applications rapidly grew awkward on machines with 4 MB of memory that could only be addressed in 64-KB pieces, requiring unnatural acts to break up simple programs into pieces that would fit. Although the VAX-11/780 was not much faster than the PDP-ll/70, the increased address space was such a major improvement that it essentially ended the evolution of highend PDP-11s. In discussing the VAX-11/780, William Strecker wrote, "For many purposes, the 65-Kbyte virtual address space typically provided on minicomputers such as the PDP-11 has not been and probably will not continue to be a severe limitation. However, there are some applications whose programming is impractical in a 65-Kbyte virtual address space, and perhaps more importantly, others whose programming is appreciably simplified by having a large address space."[3]

What's in a Chip?

The first personal computers were built using microprocessors with integer registers that were 8 bits wide, so they were called 8-bit chips and 8-bit systems. Microprocessors with 16-bit registers went into 16-bit systems, and so on. Life was simple.

The problem is, data flows to and from those registers over pathways, or *buses*, that often are not the same width as the registers. The data bus (which carries *data* to and from external locations, such as memory chips) and the address bus (which carries the *location* of the data) may be smaller or larger than the registers, depending on design considerations. This makes it somewhat confusing to decide what to call a chip.

For example, the original IBM PC had an 8088 chip with 16-bit registers, but it had an 8-bit data bus and a 20-bit address bus. Its fraternal twin—the 8086—was the same but with a 16-bit data bus. (If the 8088 were released today, Intel would probably call it the 8086SX; the 386SX is a 32-bit chip that's similar to the 386DX, but it has a 16-bit data bus and a 24-bit address bus rather than full 32-bit buses.

Similarly, all the members of the Motorola 680x0 family have 32-bit registers. However, while the 68020, 68030, and 68040 have full 32-bit data and address buses, the original 68000 has a 16-bit data bus and a 24-bit address bus.

Currently, the only microprocessor that uses 64-bit registers is the Mips R4000. The R4000 has a 36-bit address bus, which allows it to address up to 64 gigabytes of data.

For the sake of accuracy and consistency, the main text of this report refers to microprocessor size based on the width of the microprocessors' respective internal registers, without regard to the width of their data or address buses.

—*By Kenneth M. Sheldon*

Finally, we come to microprocessors. The Intel 8086 was a l6-bit architecture and, thus, likely to fall prey to the same issues as the PDP-11. Fortunately, unlike the PDP-11, it at least provided a mechanism for explicit segment manipulation by the program. This made it possible for a single program to access more than 64 KB of data, although it took explicit action to do so. Personal computer programmers are familiar with the multiplicity of memory models, libraries, compiler flags, extenders, and other artifacts needed to deal with the issues.

The Motorola MC68000 started with a more straighforward programming model, since it offered 32-bit integers and no segmentation. However, by ignoring the high 8 bits of a 32-bit address computation, it repeated the same mistake made 15 years earlier by the IBM S/360. Once again, "clever" programmers found uses for those 8 bits, and when the MC68020 interpreted all 32 bits, programs broke. Readers may recall problems with some applications when moving from the original Macintosh to the Mac II.

Table 1. CPU Characteristics

CPU	Year Released	Size Called	ISA Characteristics: Integer Register Size (R)	ISA Characteristics: Generated User Address Size (V)	Hardware Implementation: Physical Address Size (A)	Hardware Implementation: Data Bus Size (D)
DEC PDP-11/45	1973	16	16	16*	18	32
DEC PDP-11/70	1976	16	16	16*	22	32
DEC VAX-11/780	1978	32	32	31	32	64
IBM S/360	1964	32	32	24	24	8-128
IBM S/370XA	1983	32	32	31	32	128
IBM ESA/370	1988	32	32	31*	32	128
IBM RISC System/6000	1990	32	32	32*	32	64-128
HP Precision	1986	32	32	32*	32	32-64
Intel 386DX	1985	32	32	32*	32	32
Intel 386SX	1987	32	32	32*	24	16
Intel 860	1989	64	32	32	32	64
Intel 486DX	1989	32	32	32*	32	32
Intel 486SX	1991	32	32	32*	32	32
Mips R2000	1986	32	32	31	32	32
Mips R4000	1990	64	64	40-62	36	64
Motorola 68000	1980	32	32	24	24	16
Motorola 68020	1985	32	32	32	32	32
Motorola 68030	1987	32	32	32	32	32
Motorola 68040	1990	32	32	32	32	32
Sun SPARC	1987	32	32	32	36	32-64

**These processors use some form of segmentation to obtain more bits of user address space when necessary.*

The Need for Big Computers

Two common rules of thumb are that DRAM chips get four times bigger every three years and that virtual memory usage grows by a factor of 1.5 to 2 per year.[4] Additional memory is often the cheapest and easiest solution to performance problems, but only if software can easily take advantage of it.

As the natural size of code and data reaches and then exceeds some virtual address limit, the level of programming pain increases rapidly, because programmers must use more and more unnatural restructuring. If the virtual address limit is significantly lower than the physical limit, it is especially irritating, since buying more DRAM won't do you any good. Fortunately, the virtual address limit is typically larger than the physical limit, so programs may work but perhaps run slowly. In this case, you can at least add physical memory until performance becomes adequate.

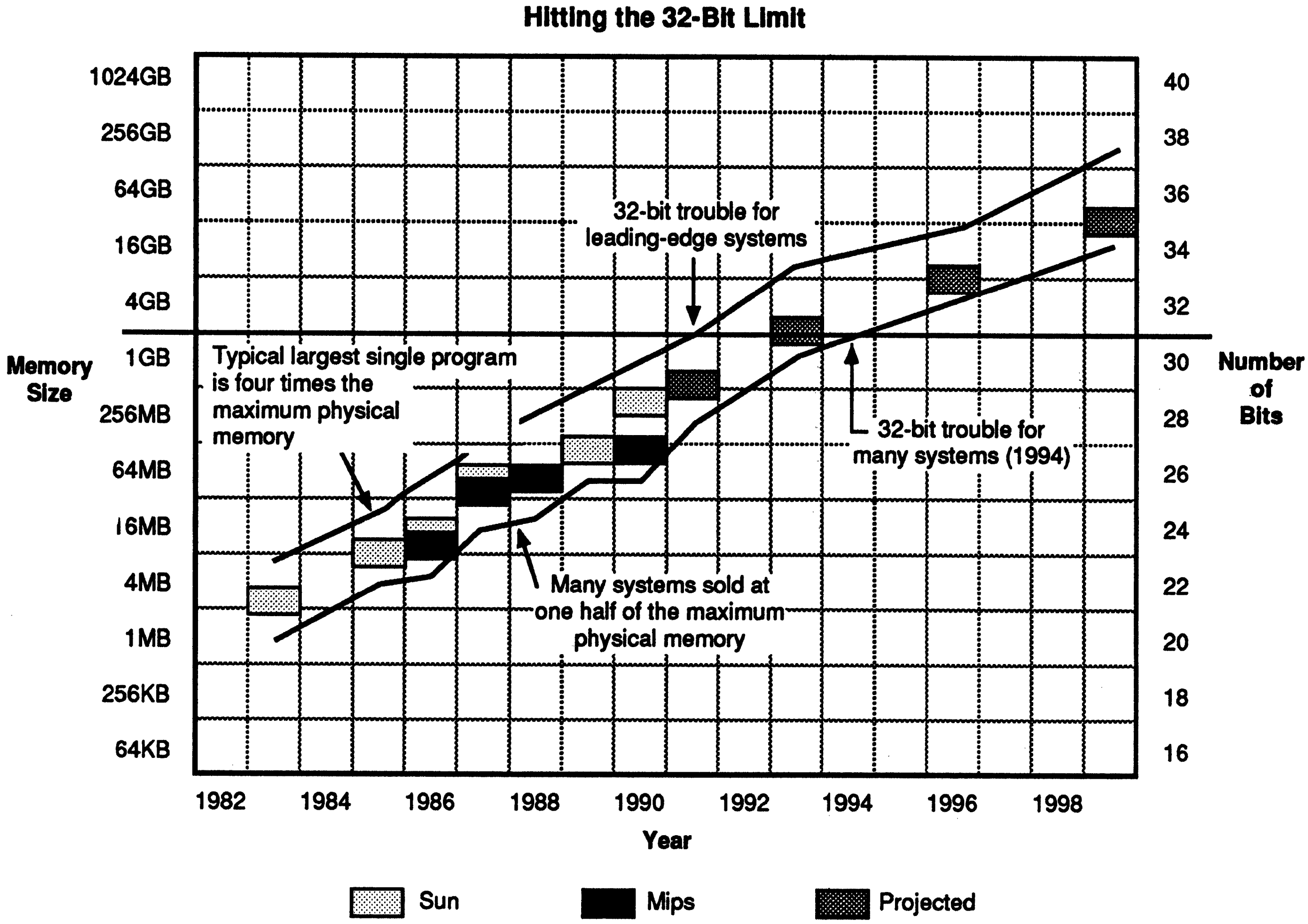
There is no definite ratio between maximum task virtual-address limit and physical address limit. Conversations with many people have convinced me that a 4-to-1 ratio is reasonable (i.e., you will actually see practical programs four times bigger than physical memory) if the operating system can support them. Some people claim that a ratio of 4 to 1 is terribly conservative and that advanced file-mapping techniques (as in Multics or Mach) use up virtual memory much faster than physical memory. Certainly, in the process of chip design and simulation at Mips Computer Systems, some of our 256-MB servers routinely run programs with virtual images that are four to eight times larger (1 to 2 gigabytes). Several companies (including Mips) already sell desktops with 128 MB of memory. With 16-Mb DRAM chips, similar designs will soon hit 512 MB—enough to have programs that could use at least 4 gigabytes of virtual memory.

32-Bit Crisis in 1993

Consider the history of microprocessor-based servers from Mips Computer Systems and Sun Microsystems. Figure 2 shows that the 32-bit limit will become an issue even for physical memory around 1993 or 1994.

As soon as 16-Mb DRAM chips are available, some microprocessors will be sold with 2 to 4 gigabytes of main memory—in fact, just by replacing memory boards in existing cabinets. You may now be convinced that Sun and Mips designers must be crazy to think of such things; but if so, they have plenty of company from others, like those at Silicon Graphics, Hewlett-Packard, and IBM. Keeping pace with DRAM growth requires appropriate CPU chips in 1991 so that tools can be debugged in 1992 and applications debugged by 1993 or 1994—barely in time.

Figure 2.
Hitting the 32-Bit Limit



Why So Much Memory?

Finally, look at the applications that put pressure on the size of virtual memory addressing. To handle virtual memory greater than 32 bits, you need either segmentation or 64-bit integer registers.

Why 64 and not something smaller, like 48? It is difficult to introduce a new architecture that runs the C language poorly. C prefers byte-addressed machines whose number of 8-bit bytes per word is a power of 2. The use of 6 bytes per word requires slow addressing hardware and breaks many C programs, so 64 is the next step after 32.

Segmentation may or may not be an acceptable solution, but there is insufficient space here to debate the relative merits. Suffice it to say that many software people with segmentation experience consider it a close encounter of a strange kind.

The following applications tend to consume virtual memory space quickly and generally prefer convenient addressing of large memory space, whether it's contiguous or sparse:

- *Databases.* Modern operating systems increasingly use file mapping, in which an entire file is directly mapped into a task's virtual memory. Since you can leave empty space for the file to grow, virtual memory is consumed much faster than physical memory. As CPUs rapidly increase their performance relative to their disk-access speeds, disk accesses are often avoided by keeping the disk blocks in large DRAM cache memories. Database managers on mainframes have long felt the pressure here, as many installations are already above 2^{40} bytes. Distributed systems designs often use some bits of the address as a system node address, with others as a per-node data address.
- *Video.* For uncompressed video, a 24-bit-color, 1280- by 1024-pixel screen needs 3.75 MB of memory. At 24 frames per second, 4 gigabytes of memory is consumed by only 45 seconds of video.
- *Images.* At 300 dots per inch, a 24-bit-color, 8½- by 11-inch page uses 25 MB, so 4 gigabytes is filled by 160 of these pages. Databases of such objects get large very quickly.
- *CAD.* CAD applications often include large networks of servers and desktops, in which the servers manage the databases and run large simulations. They naturally can make use of 64-bit software. Desktops navigate through the huge databases, and although they are not likely to map in as much data at one time as the servers, software compatibility is often desirable.

Table 2. Applicability of 64 Bits (The applicability of 64 bits differs for servers and desktop systems.)

	Server		Desktop	
Application	**Speed**	**Addressing**	**Speed**	**Compatibility**
Byte Pushing	X		X	
Graphics			X	
Big Integers	X	X	X	
Database		X		X
Video			X	
Image		X		X
CAD		X		X
GIS*		X		X
Number Crunching		X	X	

**Geographic information systems.*

- *Geographic information systems.* These systems combine maps, images, and other data and have most of the stressful characteristics of video, CAD, and GIS.
- *Traditional number crunching.* Of course, technical number crunching applications developers have never been satisfied with any memory limits on any machine that exists.

On the Desktop?

Perhaps you now believe that 64-bit servers may be reasonable, but you still wonder about the desktop. Table 2 lists the applications areas discussed, showing whether the primary use of bit systems is for speed (either in desktop or server); for addressing large amounts of data simultaneously; or for using software in a desktop system identically to its use in a server but with less actual data. Such compatibility is likely to be crucial for CAD applications but is also important for others, if only to get software development done.

For most readers, 64 bits is likely to be most important as an enabling technology to help bring powerful new applications to the desktop. The history of the computing industry, especially of personal computers, shows there is some merit to thinking ahead. Some of us remember when a 640-KB limit was considered huge.

As 64-bit systems become available, some of the number-crunching people will recompile their Fortran programs immediately, and some other developers will start working in this direction. However, I'd expect only a small fraction of applications to jump to 64 bits quickly. For example, I do not expect to see 64-bit word processors soon, if ever. As a result, an important part of 64-bit chip and software design is the ability to mix 32-bit and 64-bit programs on the same system.

Although 64-bit applications may be relatively few, some are absolutely crucial and some are indirectly important to many people. You've probably seen vendors' predictions of huge numbers of transistors per chip over the next few years. Although you may not do electrical CAD yourself, you may buy a system with those big chips; so, somewhere, people will be running programs to simulate those big chips, and those programs are huge.

I often give talks that compare computers to cars, using the CPU chip as the engine, exception handling as the brakes, and so forth. What kind of car is a 64-bit computer? Think of it as a car with four-wheel drive that you engage when necessary for better performance, but especially when faced with really tough problems, like driving up mountainsides. You wouldn't engage four-wheel drive to go to the grocery store, but when you'd need it, you'd need it very badly. Some people already have problems that require 64-bit processing, and soon more will. The necessary vehicles—64-bit microprocessors—are on the way.

References

[1]Prasad, N.S. *IBM Mainframes: Architecture and Design.* New York: McGraw-Hill, 1989.

[2]Bell, C. Gordon, and J. Craig Mudge. "The Evolution of the PDP-11." In *Computer Engineering: A DEC View of Hardware System Design,* edited by C. Gordon Bell, J. Craig Mudge, and John E. McNamara. Bedford, MA: Digital Press, 1978.

[3]Strecker, William D. "VAX-11/780: A Virtual Address Extension to the DEC PDP-11 Family." In *Computer Engineering: A DEC View of Hardware System Design.* Bedford, MA: Digital Press, 1978.

[4]Hennessy, John L., and David A. Patterson, *Computer Architecture: A Quantitative Approach.* San Mateo, CA: Morgan Kaufmann, 1990. ■

Open Systems Concepts and Issues

In this report:

Datapro Summary

Open systems is a concept embraced by both computer customers and vendors. The quest for open systems has impacted the computer industry at all levels; however, some experts believe the total "opening" of computer technology will not be achieved until the mid 1990s. Still, to accomplish the goal of true systems integration, customers and vendors together must plan for open systems today.

Executive Summary

For many years information systems (IS) customers were at the mercy of computer vendors that successfully tried to lock them into their propriety architectures. With the advent of PCs and a significant increase in IS users' level of computer proficiency, the situation is rapidly changing. The IS customer's wish list is diametrically opposite to what the vendor community has nurtured for years and includes the following:

- Integration of a new IS into the installed base of old ones
- Flexibility in selecting vendors and low switching costs if the customer decides to change
- Easy and incremental expandability of existing ISs and applications
- Graceful transition between generations of information technology (IT) that are replaced too rapidly

To resolve this conflict, customers and vendors had to find a compromise solution: the *open system.*

Open systems are not limited to operating systems or network protocols. They influence the industry progress at all levels of IS components, including software.

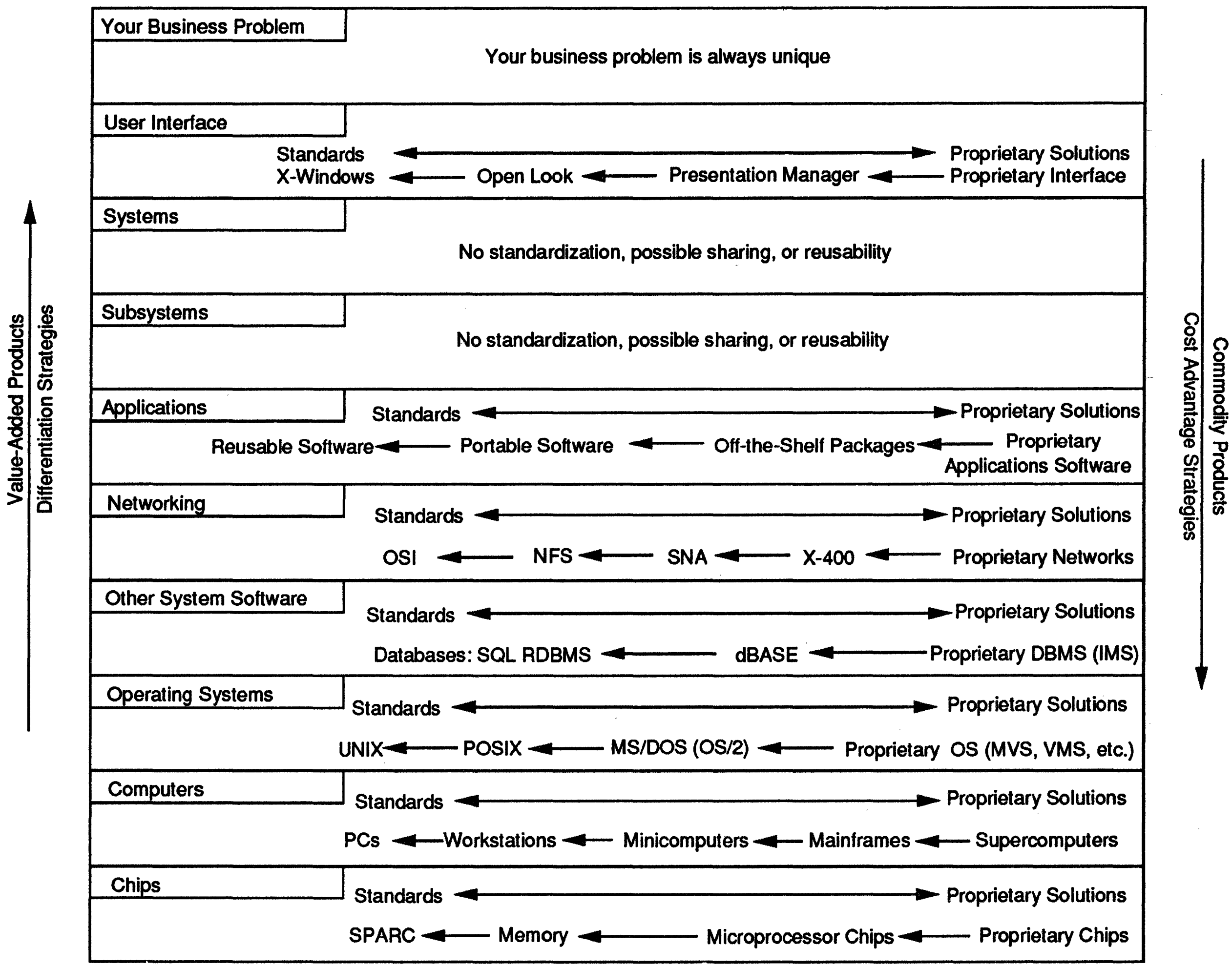
Open systems pursue the following goals: (1) standardization of increasingly higher-level IS components, turning them into commodity products; and (2) protection of the IS customer's initial investment in hardware and system application software. In other words, the center of gravity in IS projects has dramatically shifted from purchasing and putting together various pieces of hardware and system software to developing application software on top of standardized hardware/system software that is easily obtained and installed at a minimal cost.

Open systems have to be well understood because they are championed by IS customers, including United States and European governments, as well as by farsighted computer vendors. The momentum of open systems is so strong that no serious strategic information systems (SIS) project can ignore the issues discussed in this report.

What is an open system? Few IT concepts cause hotter discussions or obtain more press coverage than open systems. Several international organizations promote them, with Corporation for Open Systems, X/Open Corporation, Open Software Foundation, UNIX International, and International Standards Organization (ISO) being the most prominent.

This Datapro report is a reprint of Chapter Two, "Open Systems," pp. 41-55, from *Strategic Information Systems* by Henry Eric Firdman. Copyright © 1991 by McGraw-Hill, Inc. Reprinted with permission.

Figure 1.
Hierarchy of IS Components



In spite of this publicity (or maybe because of it), the concept of open systems is still poorly understood, and massive confusion prevails. As one example, open systems were defined in *Business Week* (May 23, 1988) as "those that rely on a common set of software that can be easily transferred from one brand of computer to another."

Does this mean that Fortran-77 is an open system just because one can write a program in Fortran-77 on DEC VAX 11/780 and then "easily transfer" it to Cray-2? Are there any differences, then, between open systems and portable software?

Why is "a common set of hardware" not included in the definition? For instance, if an Intel 80386 chip "can be easily transferred" from Compaq's Deskpro to Sun Microsystems' 386i, why don't we call this chip an open system?

Questions like these abound. In this report, I will develop a general and, hopefully, unifying concept of an open system, looking first at the conflict between vendors and IS customers as a natural source of open system movement. I will then define an open system and illustrate the concept at different levels of SIS hierarchy. I will proceed with the discussion of potential advantages of open systems and political obstacles to their proliferation. Finally, I will recommend to IS customers some plausible options for now and the foreseeable future.

IS Customers Against Vendors: An Intrinsic Conflict

One side effect of the IS industry progress over the years has been the growing conflict between hardware/software vendors and their customers. In order to have a growing and stable business, vendors want the following:

- To encourage their customers to go as much as possible for a single-vendor environment
- To lock the customer into the single-vendor solution and thus increase switching costs for the customer and entry barriers for the vendor's competitors
- To expand the customer's existing ISs and applications by offering more hardware and software
- To provide transition between generations of IT through upgrading the customers installed base and thus further locking the customer into a single-vendor environment

As recently as the early 1980s, customers had practically no other choice than to be at the mercy of computer vendors. The breakthrough came in the mid-1980s. The same microprocessor chips became available to many startup companies; competition in the vendor community increased significantly; and customers realized that, with many more choices available, they no longer had to be tied to one vendor's proprietary architecture.

With help from such farsighted vendors as Sun Microsystems, Inc., the customer community formulated its wish list for future ISs and became convinced that it is realistic and implementable. The wish list includes:

- Integration of a new IS into the installed base of old ones
- Flexibility in selecting vendors and low switching costs if the customer decides to change one
- Easy and incremental expandability of existing ISs and applications
- Graceful transition between IT generations that are replaced too rapidly

One can easily see that the customers' wish list is diametrically opposite to what the vendor community has nurtured for years. This conflict is intrinsic to the nature of the computer industry and information systems, and the only way to resolve it is to find a compromise solution equally acceptable to both vendors and their customers. This compromise solution is what I call an *open system.*

The compromise solution, however, is by no means stable. It changes in time, and at any given time is different for different vendors. For example, the compromise solution at IBM or DEC would be very different from the one at NCR or Sun Microsystems.

Thus, the first important feature of an open system is that it is a moving target. Rather than waiting for the advent of a perfectly open system, the customer community has to take advantage of what is available now and will undoubtedly be available a few years from now. At the same time, customers should become more active and influence the course of open system events.

Hierarchy of IS Components

The second important feature of an open system is that it is not limited to only operating systems or any other IS component. On the contrary, the open system philosophy permeates all nine hierarchical levels of IS implementation:

- Chips
- Computers
- Operating systems
- Other system software (database management systems, languages, development and performance evaluation tools, etc.)
- Networks
- Software applications (mostly isolated programs)
- Subsystems (software solving a specific facet of the business problem)
- Systems (functional part of entire IS)
- User interface (presentation part of entire IS)

This hierarchy is depicted in Figure 1 with the tenth level added, the level of a business problem that you want to solve in the first place and that is always unique.

Proprietary Solutions, de Facto Standards, and Open Systems

Let us look at hierarchical levels in Figure 1 in more detail. At the chip level is a spectrum of potential solutions with four points of main interest (listed in increasing order of standardization):

- Proprietary chips, such as reduced instruction set chips (RISC) developed by IBM for RS/6000 or by Hewlett-Packard for its Vectra family.
- Microprocessor chips, such as an Intel 80386 or a Motorola 68030. Many different companies can build their computers based on such a chip. (To name a few, IBM and Compaq use the 80386, while Apple uses the 68030.) Thus, microprocessor chips are less proprietary than proprietary chips. However, a certain chip, such as the 80386, is produced, or at least controlled, by only one company.

Using a microprocessor chip sometimes (but not always) implies using a certain operating system. For instance, the 68000 series is UNIX-oriented, while the 80000 series is MS/DOS-oriented. However, some 80386 computers also run under Xenix, a PC-based version of UNIX.

- Memory chips that can be used in any kind of computer—PC, workstation, or mainframe. Since there are several vendors of memory chips, they are less proprietary than microprocessor chips.
- Scalable Processor Architecture (SPARC) a RISC-based architecture proposed by Sun Microsystems that can be implemented in various configurations and semiconductor technologies. At the time of this writing, eight semiconductor companies produce different versions of SPARC, and many SPARC-based systems have been developed or are under development.

Of the four chip solutions, only SPARC deserves to be called a truly open system. (In a somewhat trivial sense, memory is also an open system but by virtue of homogeneous memory structure.) SPARC does not lock its customers, computer vendors, in one company's product and provides potentially easy and unlimited expandability and graceful scalability (transition between generations of semiconductor and computer technologies).

At the computer level, supercomputers, mainframes, and minicomputers have traditionally had proprietary architectures, although IBM mainframe clone makers make this statement not so absolute. Open systems can best be illustrated in the PC and workstation segments of the computer industry.

In the PC segment, Apple Computer's Macintosh is a typical example of a proprietary architecture, while an IBM PC is an open system: an IBM PC customer can choose among a myriad of plug-compatible PC clones, add-on boards, and, most important, a huge portfolio (more than 15,000 application packages and still growing) of application software. With its PS/2 family of PCs, IBM has been trying with some success to "close" an open IBM PC architecture.

In the workstation segment the situation is not so clear-cut. The workstation market is much more fragmented, and open workstation architectures practically do not exist. (One exception is Sun Microsystems' Sparcstation architecture cloned by Soulborne, Inc.) In the workstation segment, open system issues are passed on to the level of operating systems, where UNIX is an almost unchallenged candidate for a standard.

At the level of operating systems, there is a spectrum of more or less proprietary solutions, with four points of main interest (listed in increasing order of standardization):

- Proprietary operating systems, such as IBM's MVS and DEC's VMS.
- De facto standards such as MS/DOS, which runs millions of PCs produced by different vendors and tens of thousands of software applications also developed by different vendors. However, there is a big difference between a de facto standard such as MS/DOS and an open system: MS/DOS is still one company's product, and an MS/DOS user cannot port MS/DOS applications to computers other than the IBM PC and plug-compatibles. For the same reason, OS/2 will never become an open system even if millions of OS/2 copies are sold. Obviously, de facto standards are every vendors dream as both Microsoft and Bill Gates prove.
- IEEE 1003.1 POSIX standard, which is actually a toolkit of interface specifications rather than a complete operating system standard.
- UNIX, which currently is a de facto standard but has a good chance to become an open system, especially if Open Software Foundation (OSF) and UNIX International come to an agreement, which seems unlikely. Unfortunately, the famous line "A good thing about standards is that there are many of them to choose from" still works.

The level of other system software is represented by DBMSs. One can see three levels of standardization:

- Proprietary DBMSs, such as IBM's IMS or Cullinet's IDMS.
- De facto standards, such as dBase III. Note again the difference between a de facto standard and an open system: dBase III is one company's product (although there are dBase III clones), and a dBase III customer can neither port dBase III applications nor move data in dBase III format to other DBMSs.
- SQL-based relational DBMSs. Structured Query Language (SQL) is a standard, and any SQL-based application can theoretically be ported to any SQL-based DBMS. However, the SQL standard is very weak and, as a result, each DBMS vendor has its own SQL "standard." For example, Oracle SQL applications will run on any Oracle DBMS, from a PC-based to a mainframe-based, but will not necessarily run on other vendors' SQL-based DBMSs. Considering the trend, one can argue that an SQL-based relational DBMS will become an open system in a few years.

At the level of networking, there is again a spectrum of available solutions with four points of interest:

- Proprietary networks, such as DECNet.
- Partial standards, such as the standard for file transfer, access, and management (FTAM) or electronic mail (X.400).
- De facto standards, such as IBM's System Network Architecture (SNA) or Sun Microsystems' Network File System (NFS). SNA, which has been installed in more than 30,000 sites all over the world, is a typical de facto standard that every vendor and customer must reckon with. NFS has all chances to become an open system for additional distributed processing capabilities. At the time of this writing, more than 250 vendors and universities worldwide have licensed the NFS protocols.
- OSI, an international standard currently under development. Once completely developed, it will become an open system at the level of networking. However, at the time of this writing, OSI is no more than a set of guidelines and protocols. Two subsets of OSI, each conforming to the OSI standard, may not work together. To avoid this problem, the United States Government has developed GOSIP, a set of guidelines for determining the compatibility of OSI subsets.

At the level of applications, the establishment of a genuine open system is so difficult that it is mostly an unsolved research problem. Currently, we can see four levels of standardization:

- Proprietary application software. Today, almost all application software is of this kind.
- Off-the-shelf application packages. For some widely used applications, such as payroll and accounting, off-the-shelf application packages from several vendors are available; however, some package customization is almost always required.
- Portable software. Establishing an open system at the level of operating systems and DBMSs opens the possibility to port application software from one computer to another. However, software portability does not make an open system because application software would still be one company's product. Some portable software can achieve a status of de facto standard, as the number of its copies sold exceeds a critical mass.
- Reusable software. With the ability to be used across different, usually unforeseen, applications written by different software developers in different languages, and run under different operating systems and on different computers, software will become a genuine open system. Unfortunately, we are still a long way from the time when reusable software is widely available.

Finally, at the level of user interface, we can see three levels of standardization:

- Proprietary interfaces, such as the Apple Macintosh interface.
- De facto standards, such as IBM's Presentation Manager or Microsoft Windows.
- X-Windows, a de facto standard developed by Massachusetts Institute of Technology (MIT) and approved and adopted by scores of computer vendors. X-Windows will become an open system, but in a sense specific for user interfaces. Since the user interface is as unique as a business problem, an open system at the level of user interface is a toolkit that can be used to build an application-specific user interface.

From this brief overview, you can see that an open system is not a well defined concept. It is an almost religious movement that pursues the following common goal:

To standardize increasingly higher-level system components and thus turn them into commodity products.

This goal may be pursued on a variety of hierarchical levels. Understanding the chart in Figure 1 that represents this hierarchy is strategically important for both vendors and customers of ISs. From both perspectives an open system is the highest hierarchical level in Figure 1 at which the customer is not yet locked into the vendor solution.

Open systems do (or will soon) bring standardization to at least five lowest levels; that is, customers will be able to obtain the same hardware, system software, and networks from many vendors at the lowest possible price because open systems turn them into commodities. If this happens, how will customers be able to differentiate their SISs from the other SISs to gain a competitive advantage?

To answer this question, let us look one more time at Figure 1. Its top hierarchical level states that "your business problem is always unique." You can see that there is a break of continuity somewhere between the top level, where no standardization is possible, and the five lowest levels, at which standardization is not only possible, but has been or will soon be done.

Thus, the trend in IS development has shifted dramatically from purchasing and putting together various pieces of hardware and system software as a major part of the project, then developing proprietary application software, to developing the application software as a major part of the project on top of standardized hardware/system software that is easily obtained and installed at a minimal cost.

The important conclusion is: To gain a competitive advantage through a strategic use of IT, the company has to:

- Secure highly productive application software development staff, whether by hiring and nurturing it inhouse or by carefully selecting and closely working with an outside contractor
- Supply the application software development staff with the newest design methodologies and the best possible development tools to reduce the development life cycle and thus retain a sustainable competitive advantage for a longer period of time.

Without these two components, all your good intentions to gain a competitive advantage through an SIS will likely be futile.

The next two sections discuss the reasons why both IS customers and vendors should support the open system movement.

Why Open Systems? The Customer's Viewpoint

Why should you embrace the concept of open systems if you are an IS customer? The short answer is: "Because at any point in the customer IS's life cycle, open systems provide the closest fit with the customer wish list given previously ("IS Customers Against Vendors: An Intrinsic Conflict")," Let us discuss this answer in more detail.

The first major advantage of open systems for customers is that they get more certainty in planning the development of largescale systems, such as SISs, and migration among successive generations of technology. This certainty is especially important in the case of SISs because the SIS life cycle may encompass 5 to 15 years (or even more; consider American Airlines' Sabre first developed in 1961) so that protection of an initial investment in hardware and system and application software may be of paramount importance.

The SIS customer may be fairly confident that this investment is protected all the way up to the highest level in the hierarchy of IS components at which the system is still open. This confidence is based on a well-defined path of updates and upgrades of, or additions to, existing hardware and software, including even a complete switch from one vendor to another.

The second major advantage of open systems is that switching costs are greatly reduced so that replacement of one vendor with another will be relatively easy, inexpensive, and certainly not as painful as it is today. In fact, with the advent of open systems, customers will never again be locked into one vendor's proprietary environment.

Two immediate consequences will result when customers are no longer at the mercy of a single vendor:

- The level of customer service by the vendor will certainly go up. The open systems market is permanently a buyer's market, and vendors will fight for the customer.
- The price of hardware and software products at the open system level will inevitably go down as these products gradually turn into commodities and competition among vendors becomes more ferocious. At low hierarchical levels in Figure 1, such as PCs, we will be witnessing price wars a la McDonalds vs. Burger King.

Suppose, however, that for one reason or another, the customer does not want to switch to another vendor. The customer is a winner anyway. The reason is that multivendor environments are a political reality as well as an economically and technically justified necessity rather than a historical mishap. Customers will incrementally enhance their computational resources and software in a cost-effective way without having to marry a single vendor or to be concerned about such things as compatibility, portability, and the like. Customers will also be able to provide better utilization of their IS resources through network computing, that is, execution of application software by the set of dynamically allocated, currently free or underutilized, and most cost-effective computer network nodes.

The third major advantage of open systems for customers is reduction of customized software development costs. The reduction comes from at least two sources:

- Hiring and training expenditures will go down. A major pool of programmers will develop their application programs under a standard operating system, against standard databases, and, in the more distant future. They will also take advantage of huge libraries of standard reusable application software. They will learn all of these standards in a college, and the amount of additional education and training required will be insignificant.
- Program development productivity will go up, while quality assurance, support. and maintenance expenditures will go down as a result of standardization of operating systems, other systems software, and especially application software and user interface.

Why Open Systems? The Vendor's Viewpoint

Customers seem to be sure winners in the open systems game. Unfortunately for vendors, this statement cannot be said about all of them. Some will win, while others will lose—and lose big. The stakes for vendors are very high. Thus, it comes as no surprise that the game will be highly politicized, and political consideration will often prevail over common sense and logic. Although virtually vendors claim that they promote open systems for the sake of their customers, that claim may not always he true. Let us look at the current situation in the computer vendor camp and some plausible future alternatives.

Vendors are not created equal, and open systems are a great leveler. In other words, first-tier vendors that have established their position at the top by selling highly proprietary hardware and system software must oppose or at least resist the open systems movement no matter what they publicly say. IBM and DEC fall into this category.

You might think that first-tier vendors should sincerely support open systems because open systems turn hardware and system software into commodity products. Don't giants have a better chance to survive and prosper in the commodity market because of economies of scale and their unique ability to provide low-cost, high-volume manufacturing? If so, they could have blown away their less fortunate competitors because of their cost advantage and hence should have been pleased with open systems.

The catch is that neither IBM nor DEC wants to be in the commodity market. To win it would be a Pyrrhic victory because they should have forgotten their handsome profit margins once and forever. From the profit margin viewpoint, open systems threaten to turn first-tier computer vendors into department stores such as Sears or Kmart.

As a result, first-tier computer vendors have no other choice than to resist or at least delay open systems as much as they can. Unfortunately for them, they can exert only a limited influence on the market. For example, it would be politically suicidal for IBM and DEC to put their proprietary architectures head-on against open systems, as if to say publicly that they want to keep holding their customers the hostages of proprietary hardware and systems software.

The second best alternative for first-tier computer vendors after proprietary hardware and system software is standards based on their products or de facto standards. A huge installed base of products, new architectural initiatives such as IBM's SAA, and such organizations as Open Software Foundation (both IBM and DEC are its founding members) all provide this alternative for IBM and DEC. Speaking of Open Software Foundation, one must clearly understand that if such archrivals as IBM and DEC are doing something together, there must be a very strong reason. In this case the reason is an attempt to control (and perhaps to delay) the open systems process at the level of system software and user interface.

Unlike IBM and DEC, second-tier computer vendors such as Unisys, NCR, Honeywell Bull, and Olivetti have no other choice than to vehemently support open systems. As mentioned previously, open systems are a great leveler, and the only way these companies can compete with and take some market share back from IBM and DEC is to promote open systems actively.

For the same reason, workstation vendors such as Sun Microsystems and Hewlett-Packard (HP) actively support open systems. Sun Microsystems has even become an initiator of the open system movement at the level of chips, operating systems, networks, and user interfaces.

Of course, second-tier computer and workstation vendors clearly understand that open systems lead to a commodity-like computer market. They also understand that open systems will result in lower prices and profit margins. Thus, their only hope is further penetration of computers in all areas of our life and business and, as a result, higher sales in their respective markets.

It is hard to say if such hope is completely justified. On one hand, Sun Microsystems, one of the most successful and thriving computer companies in history, is at the same time a company with the lowest profit margins in the computer industry. On the other hand, there may not be enough room for another dozen companies, some of them significantly larger than Sun. It is relatively easy to predict a bloody war at the law end of the workstation market, where Sun may be challenged not so much by IBM, DEC, or HP as by Apple and Compaq.

As follows from Figure 1, the only way to preserve today's margins is to go up in IS component hierarchy; that is, get involved in the application software and eventually system development and integration business. That is why IBM, DEC, and AT&T are trying to become major players in the system integration business, while HP, Sun Microsystems, and Apple demonstrate their presence in it.

Having hundreds of dedicated third-party software vendors, as Sun currently has, may no longer be enough. With the advent of open systems, third-party software vendors may also be dedicated to any other open systems vendor. By the nature of an open system, third-party software will run on any hardware that supports open system standards.

Who is Pushing Open Systems?

The open systems movement has several powerful supporters. First, the hardware and system software vendors have a vested interest in open systems. As mentioned in the previous section, such companies as Sun Microsystems, AT&T, Unisys, NCR, and HP actively support open systems.

Second, United States and European governments actively support open systems. The U.S. Government, a $17 billion-a-year IS user, actively promotes UNIX and OSI as an open systems solution at the level of operating systems and networks, respectively. The European Commission already requires open system software for all government computer bids.

Third, some very big IS users and system integrators, such as Boeing and General Motors, really want standards and support proliferation of open system standards as well as actively participate in their creation. Strangely enough, many IS users are fairly indifferent to open systems. Unless IS users become active and say what their requirements are and in what direction the open systems movement should go, they will get only a limited edition of open systems: open systems, vendor style. And it may not be the same as open systems, customer style.

Here are the major reasons—some of them well justified—why IS users still play a relatively passive role in the open system movement:

- Many users remain loyal to their vendors, often for a good reason.
- The largest installed base of business are IBM sites, where users are more conservative and pick a "wait-and-see" tactic.
- Open systems are not on the top of their priority list (for example, many users are more concerned about connectivity).
- Immediate conversion from proprietary architectures to open systems is prohibitively expensive and economically unjustified.

- Many users are concerned that standardization implies reducing the level of technology to the least common denominator.
- Some major standards are claimed and talked about a lot, but not really implemented.

Problems with Open Systems

Two problems with open systems are commonly cited by their opponents:

- Open systems compromise security because they offer greater portability, interoperability, scalability, and information sharing. OSF defines portability, interoperability, and scalability as follows:
  - –Portability: the ability to run application software on computers from different vendors.
  - –Interoperability: the ability for computers from different vendors to work together.
  - –Scalability: the ability to host the same software environment on a wide range of computer platforms, from PCs to supercomputers.
- Open systems bring standardization and thus impede innovation because innovation is always deflection from standards.

Let us briefly discuss these problems.

Open systems and security

Greater portability, interoperability, scalability, and information sharing should be balanced with security requirements. Unfortunately, these elements are not always balanced properly, and security breaches result. The most common violations of security include:

- Porting uncertified software that may turn out to be infected
- Sending private or sensitive information across the computer network without necessary precautions, such as encryption
- Providing weak protection of computers and the network from the spread of viruses
- Allowing too much freedom in accessing privileged information and programs

Historically, security has had a lower priority than other technological developments, especially in UNIX environments, a major stronghold of open systems. For example, in a recent survey of UNIX users, security was not even mentioned as a problem, although UNIX is not known as the least vulnerable operating system.

Indeed, the fact of life is that open systems have increased vulnerability. However, treating security as one of the major IS design considerations can eliminate many problems that should not have occurred in the first place. Developing an open system first and only then providing its security is a common and serious mistake that can be easily avoided.

Another common mistake is to consider security as a component rather than a system problem. The most secure operating system will not help much if security measures are not provided and enforced at the database level; nor will the secure operating system with effective security measures help in a distributed system that has an insecure computer network.

Open systems and innovation

Since open systems pursue standardization, the conflict between open systems and innovation is intrinsic and inevitable. It may be resolved only by looking at open systems as a dynamic rather than static process.

Standards always lag behind an advanced technology and, unfortunately, usually catch up with it at the time it becomes mature, if not obsolete. For SISs that are designed to gain a competitive advantage, the time factor in using the advanced technology is so critical that the existing standards may be completely ignored just because they are still immature. By the time standards become mature, the strategic use of the specific technology is an open secret. Similarly, high-payoff applications are more often ad hoc and proprietary than standardized.

The resolution of this conflict between standardization and innovation comes from understanding that the most innovative and strategically important systems still have to be developed on top of standards in the areas that are well established and not crucial for innovation. Conversely, today's innovation is tomorrow's standard, provided that the innovation has proved itself successful. These two processes must coexist, complement each other, and be well managed if the enterprise is to reap benefits of standardization and innovation at the same time.

Conclusion

From this reporter you may have gotten overoptimistic expectations that open systems are available today, or at least will appear very soon, maybe tomorrow. Such expectations are naive. First, open systems will not happen overnight. Think of open systems as a process of "opening" information systems—a process that is intrinsically gradual, incremental, and in some cases even painful.

Second, this process is highly politicized by its many participants, who may have different vested interests in how it will proceed, what results it will have, and when these results will occur. IS users, major beneficiaries of open systems, must play an active role in this process if they are to gain all the advantages of open systems.

Third, in order for open systems to materialize, standards at different hierarchical levels must be implemented rather than only claimed or discussed. Many standards for open systems should be discussed and agreed upon in meetings of international standards organizations, such as the ISO or Comite Consultatif International de Tèlègraphique et Tèlèphonique (CCITT). This process automatically means years of discussion; thus both the vendor and user communities are doomed forever to working with incomplete or temporary standards.

As a result of these factors, some skeptics do not even expect radical changes in the implementation of open systems until the next century (which is not that distant). My optimistic view is that we will see implemented open systems by 1995. In order to accomplish this goal, however, IS customers have to be prepared today. ■

Interoperability Today

In this report:

Datapro Summary

The concept of interoperability is one of the most important in computing, but the term has been used so much that its meaning has become blurred. Interoperability is a matter of making LANs, computers, and computer components communicate. It is closely linked to the existence of accepted standards, both formal and de facto. From the standpoint of interoperability, the most important standards are those defined in the OSI Reference Model. Interoperability is well defined at the lower levels of the Model, but it is less well defined in the upper layers. This report provides examples of the state of interoperability at each of the seven OSI layers, describing the degree of standards certainty at each layer.

The more you encounter a word or phrase, the less meaning it begins to have. In computers, you see *user-friendly* and *powerful* so often that the eye passes over them with barely a whisper reaching the brain. Lately, I've been tempted to add *interoperability* to the list; it's become overworked and misused. Yet it is one of the most important concepts in computing. It's time to put some meaning and substance back into *interoperability* to help us talk meaningfully about what works with what.

The biggest problem is that interoperability is a huge concept; you can use it in regard to nearly everything. You most often encounter it in reference to LANs (and most of my examples below deal with LANs), but it also touches on other ways that you can make computer systems and components talk to one other. Modems, serial ports, printers, printer ports, E-mail links, applications software, musical instruments, telephones, and even fax machines exemplify varying degrees of interoperability. The challenge is to define a framework in which an examination of interoperability makes sense.

This Datapro report is a reprint of "Interoperability Today" by Barry Nance, pp. 187-196, from *Byte*, Volume 16, Number 12, November 1991. Copyright © 1991 by McGraw-Hill, Inc. Reprinted with permission.

Interoperability and Computers

One of the most interesting characteristics of *interoperability* is that you no longer use the word in reference to things that actually do interoperate. If you've ever purchased a telephone and some phone cord with RJ-II jacks at either end, plugged the phone into a wall jack in your house, and immediately picked up the handset to make a call, you've experienced interoperability of the highest order. But you'd never use the word to describe to your friends what you did. The same is true of every electrical appliance you've ever plugged into a wall outlet. The degree of interoperability achieved by telephone and utility companies is exceedingly high.

On the other hand, if you've ever tried to merge two LANs, one using IBM OS/2 LAN Server and one using Novell NetWare (and perhaps with some Macintosh workstations attached), so that all users get to share the same resources and files on all the servers, you know how far the computer industry has to go before it achieves true interoperability. In such a situation, you might find yourself using *interoperable* as an expletive while you try to make things work.

If you buy all your hardware and software from one manufacturer, your concern for interoperability is eased somewhat, although you can still run into things that don't work well together. For example, IBM produces charts and tables to show which of its products work together and, by implication, which do not. And you can't dismiss IBM's proprietary products from interoperability considerations; IBM is large enough that its specifications and guidelines often become industry standards. Interoperability is closely linked to the existence of accepted standards, whether they're the formal result of official standards efforts or the de facto result of industry-wide acceptance.

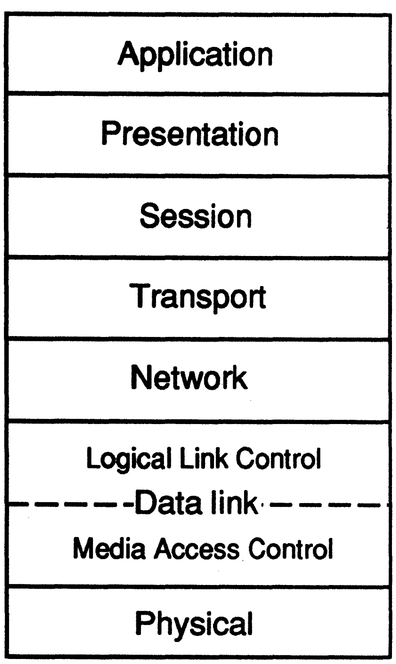
Up the Stack

The most important standards in the interoperability arena arc those encompassed by the Open Systems Interconnection Reference Model. It provides a framework that developers use to create interoperable products and that you can use to explore the current state of interoperability. Because many products are not yet OSI compliant, the correspondence between the OSI model and reality is not always exact, but it is the best reference available.

Briefly, the OSI model declares seven protocol layers and specifies that each be insulated from the others by a well-defined interface. From bottom to top, the seven layers (see Figure 1) are as follows:

- *Physical.* The lowest level of the OSI model specifies the physical and electrical characteristics of the connections that make up a network. It encompasses such things as twisted-pair cables, fiber-optic cables, coaxial cables, connectors, and repeaters. You can think of it as the hardware layer.

Figure 1.
The OSI Model



Each layer of the OSI model presents a strictly defined interface to the adjacent layers, permitting pieces from different vendors to fit together seamlessly.

- *Data link.* In this layer of processing, the electrical signals enter or leave the network cable. Bit patterns, encoding methods, and tokens are examples of elements known to this layer (and only to this layer). The data-link layer detects errors and corrects them by requesting retransmissions of corrupted messages or packets. Because of its complexity, the data-link layer is broken into a Media Access Control layer and a Logical Link Control layer. The MAC layer manages network access (e.g., token passing or collision sensing) and network control. The LLC layer, operating at a higher level, sends and receives the data messages or packets.
- *Network.* This layer switches and routes the messages to get them to their destinations. It is responsible for addressing and delivering messages.
- *Transport.* When more than one packet is in process at any one time, this layer controls the sequencing of the message components and regulates inbound traffic flow. If a duplicate packet arrives, this layer recognizes and discards it.
- *Session.* The functions that are defined in this layer let applications running at two workstations coordinate their communications into a single session. A session is an exchange of messages—a dialogue—between two workstations. This layer helps create the session, informs one workstation if the other has dropped out of the session, and terminates the session on request.
- *Presentation.* When IBM, DEC, Apple, Next, and Burroughs computers all want to talk to each other, some translation and byte reordering is needed. This layer converts data into (or from) a machine's native internal numeric format.
- *Application.* This is the layer an application program—and, therefore, the programmer—sees. A message to be sent across the network enters the OSI model protocol stack at this point, travels downward toward the first layer (the physical layer), zips across to the other workstation, and then travels back up the stack until it reaches the application on the other computer through that computer's own application layer.

What follows is a discussion of the current state of interoperability at each of the layers in the OSI stack. Notice that while the status of interoperability gets fuzzier as you go up the stack, the word *interoperability* is used (or perhaps abused) more frequently as you near the top.

Firm Foundations

Within each type of cable, physical interoperability is pretty well defined. If you indicate you've connected some computers with RG-58 A/U coaxial cable, there is a good chance that a network administrator will immediately think of thin-wire Ethernet. Mention shielded twisted-pair (IBM Type 1) cable, and Token Ring will probably come to mind. Unshielded twisted-pair cable might be Token Ring, or it might be Ethernet's relatively new version (10Base-T). Fiber optics? Fiber Distributed Data Interface (FDDI) is new enough that you might just get puzzled looks.

In each case, though, the simple mention of the type of cable is often enough to define the entire physical appearance of the network components and to specify what will connect—interoperate—with what. If my computer has a Token Ring card and I want to connect to your network, I need ask only two questions to find out whether it's possible: "4 or 16 Mbps?" and "Shielded or unshielded twisted pair?" This level of interoperability is made possible by standards efforts managed by the IEEE. The 802.3 (Ethernet) and 802.5 (Token Ring) standards specify exactly how the physical layer of the network should behave, and these standards extend their reach into the data-link layer.

There are other physical standards that are highly interoperable. The best example is a parallel printer cable (which uses the Centronics interface). If you buy a printer with a parallel interface and it doesn't work with a computer and cable known to be good, it's certain that the printer is dead on arrival.

The standard for serial cables, RS-232, is equally rigorous and exacting. It defines Data Terminal Equipment and Data Communications Equipment, and it specifies exactly how to connect DTE and DCE to make them interoperate. I can buy a modem with an RS-232 interface, connect it to my RS-232 serial port with an RS-232 serial cable, and know that things are going to work.

Data Link: Defining Data Formats

Suppose I got out some spare chips and a soldering iron and somehow managed to connect my Token Ring-equipped computer to your Ethernet LAN so that the Ethernet message packets (*frames*) successfully entered my Token Ring card (don't try this at home!). Can I "interoperate" on your LAN? No way.

First of all, my Token Ring adapter needs to see a 3-byte *token* that it can claim before it can send data on the network. The cards in the other workstations rely on collision sensing to get their message across. But even forgetting about tokens and collision sensing, the basic format of the data is different.

Pascal programmers call it a RECORD. In C, it's a struct. Bare-metal assembly language folks might use a struc <>. COBOL programmers think of it in terms of a Record Description. Basically, it's the layout of the data in memory, the organization of the data bytes into fields within a data record.

Not only is an Ethernet data record or frame laid out differently from a Token Ring frame, IEEE 802.3 Ethernet is also slightly different from *true Ethernet*—the original Ethernet definition from Xerox and DEC. Figure 2 shows the differences. But within each standard (IEEE 802.3 Ethernet or IEEE 802.5 Token Ring), the degree of interoperability is high, permitting adapter cards of the same type from different manufacturers to interoperate seamlessly.

ARCnet data records (or frames), FDDI frames, and StarLAN frames are differently laid out, too—just as you'd expect. The essential point, though, is that my hybrid, jury-rigged Token Ring card won't work with your Ethernet LAN because these record layouts are understood by the ROM program code burned into each adapter. (If I did manage to modify the Token Ring card enough to make it work, it would no longer be a Token Ring card—it would have become the world's most expensive Ethernet adapter.)

What about non-LAN examples? If interoperability is such a slippery thing, why can I use my fax machine to send documents and pictures to you as if we had picked our equipment from the same assembly line? The answer, of course, is standards. Group 3 fax is a standard understood around the world.

The 9000-bps modems took a while to catch on, mostly because the standard for data representation was still evolving. Now we have CCITT V.32 modems from different manufacturers that reach out and touch each other reliably. However, some modems, designed before the CCITT standard stabilized, do not always speak clearly to other 9000-bps modems. An example is the USRobotics Courier HST modems; before USRobotics' Dual Standard, its modems didn't interoperate well with other 9600-bps modems. (The original HST used a proprietary modulation scheme that provided a 9600-bps forward channel and a back channel running at 1200 bps.)

Network and Transport Layers

People talk a lot about interoperability at the network and transport layers. If my earlier cynical observation is true, this means that interoperability is sporadic and elusive at this level. I'll test this theory here.

IPX.COM, the layer of NetWare workstation software between NETx.COM (above) and network adapter software (below), exists at this level. IPX has its own application programming interface, and applications software on one NetWare workstation can talk through this interface to other NetWare workstations. However, the workstation dialogue takes place in a unique dialect of network languages; other workstations may use the same cables and network adapters, but they can't participate if they don't speak the IPX language.

Incidentally, if you are getting confused by this talk of layers, protocol stacks, and interfaces, look back at Figure 2 for a moment. See the field in the Ethernet and Token Ring frames that's labeled *Data*? The data field is yet another whole record layout.

The fields in this encapsulated data record are defined by the software in the network/transport layers. If I send an Ethernet frame to your NetWare workstation with a data field (i.e., record) containing bytes organized a certain way, your IPX-based workstation will recognize the frame. If I put incorrect or undefined values in those bytes, your workstation will not know what to do with the frame, even if the Ethernet portion is filled in correctly. At the network and transport layers, interoperability is defined mostly in terms of the definition of the data within the message packet.

IPX is a close adaptation of a protocol developed by Xerox, XNS. Novell uses IPX, of course, but I don't know of any other major players whose LAN products implement it directly. One way around this stumbling block to interoperability is to use the Clarkson Packet Drivers.

Supported by Novell, FTP Software (Wakefield, MA), much of the academic community, and other groups, the Clarkson Packet Drivers allow multiple protocols to use the same network adapter. IPX data packets are routed to IPX, NetBIOS packets to NetBIOS, TCP/IP packets to TCP/IP, Network File System packets to NFS, and so forth. These drivers take up a little extra RAM, but they do a good job of providing interoperability at the network/transport layers—if you have multiple upper layers that all need to use the same network adapter.

TCP/IP is yet another piece of the interoperability puzzle. (If everyone used TCP/IP—or IPX, or NetBIOS—I probably wouldn't be writing this.) TCP/IP was developed with government money and is therefore in the public domain; this probably explains its popularity. TCP can be considered a transport-layer definition; IP fits more into the session layer. TCP, of course, is very different from

Figure 2.
Frame Types

Different LAN types encapsulate data differently, resulting in incompatibilities. The most significant difference between "true" Ethernet and IEEE 802.3 Ethernet is the Pad field. True Ethernet defines the smallest data field as 46 bytes; 802.3 Ethernet allows you to send a zero-length data field and pads the frame to the proper length. Token Ring is something else again: It has one frame definition for the token and another for the data frame. Given the differences between the Ethernet and Token Ring frames, it's no wonder that adapters of one type can't recognize frames of the other type.

Frame Types

Ethernet frame

Preamble	Destination	Source	Type	Data	CRC	Field length, in bytes
8	6	6	2	46–1500	4	

IEEE 802.5 Ethernet frame

Preamble	SFD	Destination	Source	LEN	Data	Pad	CRC	Field length, in bytes
7	1	2 or 6	2 or 6	2	0–1500	X	4	

SFD = Start frame delimiter
Pad = filler bytes to bring the frame to a minimum length
CRC = Cyclic redundancy checksum
LEN = Length of data field

IEEE 802.5 Token Ring token

SD	AC	ED	Field length, in bytes
1	1	1	

SD = Start delimiter AC = Access control ED = End delimiter

IEEE 802.5 Token Ring data frame

SD	AC	FC	DA	SA	Data	FCS	ED	FS	Field length, in bytes
1	1	1	2 or 6	2 or 6	Variable	4	1	1	

SD = Start delimiter
AC = Access control
FC = Frame control
DA = Destination address
SA = Source address
FCS = Frame-check sequence
ED = End delimiter
FS = Frame status

IPX. In fact, the many different transport-layer definitions (record layouts and data field meanings) are the biggest obstacles to interoperability.

Another term that is used at this level is *transport independence*; it's another way of saying interoperability. Suppose you want to develop an application consisting of different modules that run on an IBM PS/2 under OS/2, a Mac, and a UNIX box. You'd like to design the modules so that the calling interface—one module passing parameters and control to another—is completely transparent to the programmer. You need special glue to put these pieces together. This glue is called *remote procedure calls*, or RPC.

RPC lets different kinds of computers process different parts of the same application. Companies like SunSoft (a subsidiary of Sun Microsystems), NetWise, Novell, Hewlett-Packard, and Momentum Software are doing amazing things with RPC. American Airlines, for example, is using XIPC from Momentum to develop a new cargo-routing system that runs on a variety of computers.

Transport independence means that all the machines making up the entire application system are using the same transport-layer definition—IPX, NetBIOS, TCP/IP, or something else. RPC is great when the same transport layer is available for all the kinds of computers you want to use. You use the RPC tools to segregate your modules by target machine, and you link the modules with an RPC library on each target. Voilà—you have taken the best characteristics of each kind of computer and incorporated them into one system.

But suppose you'd *really* like to use one of those new Volga Boatntan/2000 computers, which as everyone knows only support the F/X protocol, in your application. Your vendor of RPC tools shrugs and says, "Sorry." You start traveling to the trade shows, hoping to hear someone use the words "interoperability" and "Volga" in the same breath.

I've concentrated on IPX and TCP/IP, probably the two most widely used transport protocols. Another term you hear frequently in conjunction with TCP/IP and IPX is NetBIOS. Actually, though much of NetBIOS works at the session layer, the next level of the OSI model.

Session Layers

IPX.COM actually contains two protocols, IPX and SPX. As you'd expect, SPX is a layer on top of IPX and uses IPX to send and receive its data messages. SPX is session-oriented, like NetBIOS, but the similarity ends there. In my book *Network Programming in C* (Que Publishing, 1990), I explain NetBIOS programming techniques in one chapter and IPX/SPX techniques in another—the degree of interoperability between the two chapters is exceedingly low.

"Ah!" you say. "Novell supplies a NetBIOS.EXE program with NetWare that ought to solve the problem." I wish that it were true. If the various implementations of NetBIOS were interoperable, you could construct a protocol stack consisting of network-adapter support software, IPX.COM, NetBIOS, and different redirector modules from Novell (NetWare NETx.COM), Artisoft (LANtastic), Performance Technology (PowerLAN), IBM (DOS LAN Requester), and other vendors, and you'd be able to access just about any file server ever created.

Even if you had enough RAM to hold this protocol stack, you still couldn't use it to access anything. Each vendor expects its LAN software to work with the NetBIOS *it* implemented. I've gone only five-sevenths of the way up the OSI model, and already interoperability is out of focus and getting fuzzier by the minute.

While there are some differences in each of the programming interfaces of the NetBIOS implementations from various vendors, a bigger problem is the different data-record layouts that are used to shunt NetBIOS packets across the network. On a DOS workstation, the NetBIOS module from IBM is the device driver DXMT0MOD.SYS (part of the IBM LAN Support Program). Novell's, as I mentioned, is NETBIOS.EXE. Artisoft's NetBIOS for LANtastic is AILANBIOS. Performance Technology's PowerLAN NetBIOS module is named for the adapter it supports. The data records (i.e., the contents of the data record encapsulated inside the data field of the Ethernet or Token Ring frame) that each of these modules creates are completely different.

Presentation and Application Layers

The software on your LAN that gives you access to the file server is the network shell, or redirector. On a NetWare network, this is NETx.COM (where the *x* denotes the version of DOS you run). On an OS/2 LAN Server network, it's represented by DOS LAN Requester at DOS-based workstations. Other vendors' redirector software is usually called REDIR or NET. At this level, you can also talk about E-mail interoperability, along with the software mechanisms for sharing data in the Presentation Manager and Windows environments.

It goes without saying that NetWare LANs, OS/2 LANs, Banyan LANs, and peer-to-peer LANs are not easily interoperable at this level. Just as you'd suspect from the discussion of lower layers of the protocol stack, the lack of communication among different vendors' redirector software modules is mostly a matter of data definition.

NetWare workstations use the NetWare Core Protocol to request file services; servers respond in kind. NCP was highly proprietary until earlier this year, when Novell announced it would license the NCP specification for a fee.

IBM designed the Server Message Block protocol for use with the PC LAN Program. One of the best documents for understanding the SMB protocol is Volume 2, Number 8-1, of the *IBM Personal Computer Seminar Proceedings.* Printed in 1985, it's still a useful introduction to how SMBs are used. IBM currently uses the SMB protocol between OS/2 LAN Server and DOS LAN Requester workstations; a few vendors of peer-to-peer products (e.g., PowerLAN) also support SMBs. Like NCP, however, SMB is identified closely with one company. It doesn't have the clout to become a widespread standard.

Interprocess Communications

At the top of the OSI model, new products and new standards for application-level interprocess communications come into being so often that it's hard to get a handle on the current state of affairs. *Interoperable* is used to describe each new product and standard. As I've suggested, the frequent use of the word is perhaps significant, but not in the way that was intended.

Dynamic Data Exchange, an application-to-application protocol for the Windows environment, was designed by Microsoft for use in Excel, the spreadsheet program. It is described in the Windows Software Development Kit and is available to other applications that want to talk to each other. Microsoft also offers Object Linking and Embedding (OLE) for use under Windows. DDE has five mechanisms applications can use:

- *Hot Link.* A server application sends data to a client application whenever data changes.
- *Warm Link.* The server notifies the client that data has changed, and the client can then request it.
- *Request.* This is equivalent to a copy-and-paste operation between the server application and the client, without the need for the intermediate clipboard.
- *Poke.* This is a "back channel transfer."
- *Execute.* One application controls the execution of another.

For two applications to use DDE, they must agree on the format of the data to be exchanged. If the applications are from different vendors, this means the vendors must publish detailed specifications.

OLE is a layer on top of DDE that insulates the application programmer from some of DDE's tedious detail. One application puts data into a container that is located in the other application. The second application only needs to know how to display the data. If changes to the data are required, the second application invokes the first one through a special interface.

HP's NewWave is object-oriented. It uses the concept of agents, and it contains an intelligent link manager that resolves object names into file-system names. Version 3.1 of NewWave incorporates OLE. SunSoft's ToolTalk is another example of interprocess communications and object linking.

Interoperability among the various providers of interprocess communications facilities probably won't happen for a long time. Even within a single protocol like DDE, interoperability remains highly application dependent.

E-Mail Interoperability

E-mail is another application-level service that's feeling the interoperability push. Many commercial E-mail products adhere to the Message Handling System and/or the X.400 standard. MHS messages, for example, have a header, a body of text, and perhaps an attached file. The header contains a destination address, a return address, a postmark, and other information. Both addresses have a particular format: <username> @<workgroupname>. Applications that define data in this same format can interoperate with MHS. Similarly, applications that use X.400 conventions can interoperate with other X.400 applications and users.

Of course, myriad other E-mail standards exist. Western Union has EasyLink, Telenet offers Telemail, and a rather popular one is PROFS from IBM. Interoperability among these systems is mostly a matter of reformatting data to look the way the other system expects. In E-mail, at least, some of the promise of interoperability has been realized.

Farther on Down the Road

Interoperability is a big issue; so big, in fact, that an annual trade show, called Interop, is now devoted to it. Run by the Advanced Computing Environment people, Interop has one overriding criterion for its exhibitors: Products must successfully interoperate with other products on a network. The show is the meeting ground of the technical people who sweat bullets to turn *interoperability* into a word you won't have to hear much of anymore.

And they still have a lot to sweat over. While interoperability solutions are solidly in place for the lower levels of the OSI stack, things begin to break down around the transport and session layers. Constructing a heterogeneous network today is not an impossible undertaking, but the day when any application can interact seamlessly with any other over your network still remains at least a few years off. ■

Graphical User Interfaces (GUIs): Trends and Applications

In this report:

Datapro Summary

Graphical user interfaces (GUIs) are icon/pointer-based operating shells. They claim to offer an easy-to-use and learn, more intuitive alternative to character/function key-based interfaces, such as standard DOS and UNIX operating systems. Popularized by the success of the Apple Macintosh and revitalized by the impressive sales of Microsoft Windows 3.0 for DOS-based workstations, GUI use is increasing. But many questions about the real benefits of GUIs remain unanswered.

GUIs: The Promise

Graphical user interfaces (GUIs) were first popularized with the advent of the Apple Macintosh personal computer in 1984. The Macintosh was the first PC to incorporate a GUI as an intrinsic component of the operating system. Macintosh claimed to offer a more intuitive, easy-to-use and learn alternative to the character/function key-based interfaces provided by the DOS and UNIX operating systems.

The Macintosh's mouse-driven multi-window/file, folder/desktop metaphor captured the attention of both system managers and computer users, as well as the computer industry media. They praised the graphical approach that Apple had taken with the Macintosh, and heralded GUIs as the wave of the future. Many DOS based IBM-PC and UNIX system users were skeptical about the real benefits of GUIs at the time of the Macintosh's release. However, the popularity and acceptance of GUIs on virtually every major computing platform has steadily increased.

Elements of a GUI

As an example, we will examine the GUI of the Macintosh operating system. Since its earliest incarnation, it has contained nearly all of the elements and operational features that are considered critical to the definition of a modern GUI. Nearly every other GUI developed since provides some variation of the key elements of the Macintosh GUI.

GUIs are composed of several distinct components which work together to make it possible for users to interact with their workstations using methods that are considered to be more consistent with the way tasks are completed. By providing a more intuitively familiar means of operating the system, GUIs claim to eliminate the need for learning system-specific commands and syntax.

The Desktop

This is the underlying visual platform upon which the interaction between the user and the system takes place. It fills the entire physical screen, acting as a visual metaphor that mimics a user's actual desk or work area. The desktop includes a menu bar running across the top or down the side of the screen. The menu bar provides access to groups of the most common operating commands.

Mouse and Pointer

The mouse is a hardware peripheral that the user can physically move to control the pointer. The pointer is a small arrow which appears to float over the desktop. Depending on the system, a mouse is equipped with 1, 2, or 3 buttons. By manipulating the

—By Robert Levin

mouse in conjunction with its buttons, users can perform most computing activities without having to use a function key or type text onto a command line.

Mouse input is versatile. For example, with the pointer positioned over a data file, a single click of the mouse button can highlight, or select, the file. By highlighting and continuing to hold down the button, the file can be "dragged" to another directory in order to move or copy it, while a double click (two presses in rapid succession) on the file would access the application in which the file was created, and bring the file onscreen for user manipulation.

Windows

Windows are rectangular subdivisions of the desktop through which users gain access to applications and data files that reside on the system. GUIs allow users to open multiple windows and move, overlap, and resize them to facilitate access, and to allow maximum use of the limited amount of space available on the desktop. Within an application, each individual file or document resides in its own window. Many GUIs offer some form of multitasking, which allows users to run more than one application at a time. The GUI also makes it easier to transfer textual, numeric, and/or graphical data between applications.

When the user is not in an application, windows function as the graphical equivalent of system directories or subdirectories, with individual windows typically displaying the names of files and applications residing in a particular directory of either local storage devices or available network servers.

Icons

An icon is a small graphic representation of an application, file, storage device, or computer activity. Icons serve as a sort of object "handle," and can be used in a variety of ways to perform activities such as opening an application, copying a file, or printing a report, through various manipulations of the mouse.

The GUI combines these components into an easily learned means of communicating with the computer. Users move the mouse pointer around the desktop, selecting specific files or icons much as they would physically reach for a pen or pencil. By pointing and clicking the mouse buttons in various sequences and combinations, the user can open, move, resize, and close windows, and move files and icons. Users can start applications and perform most other computer functions in less structured, non-linear ways that are most comfortable for them.

Who is Using GUIs?

When Microsoft released Windows 3.0 in 1990, it was heralded with a promotional campaign that dwarfed any in recent memory. Whether thanks to, or in spite of, the hyperbole, GUIs have finally become a full-fledged industry buzzword. In its first year of availability, Microsoft sold nearly four million copies of Windows 3.0. Estimates place the number of DOS-based workstations that will be equipped with GUIs at nearly eight million by 1992. In addition to the millions of GUI-based Macintoshes, GUI mania is sweeping into many large-scale UNIX-based installations as well; it is estimated that over one million UNIX workstations are GUI equipped. There is little doubt that GUIs are here to stay.

Still, questions remain. What are the real benefits of GUIs? What types of users stand to gain the most from the use of a GUI? What are the hidden costs of implementing them on a system-wide basis?

The Benefits of GUIs

Ease Of Use

By eliminating the need to learn complex and lengthy system commands or divergent application-specific syntax, GUIs offer the potential for greatly increased ease of use. This is especially true in the case of users with little or no training in the operation of computers and data terminals. Both GUI proponents and opponents generally agree that for the new user, GUIs are less difficult to learn and use, requiring less memorization and attention to order of execution. If a user gets lost, or forgets a command and just does not know what to do next, the visual nature of a GUI makes it easier for the user to get back to the starting point and try again.

Many text-based interfaces provide little or no warning when a user deletes stored information. GUIs tend to be more forgiving of mistakes, often prompting the user with small dialogue-box windows that ask if they are sure they want to continue with any procedure that will eliminate or change data.

Reduced Training Costs

Another benefit of implementing a GUI is the marked decrease in the amount of time and effort required to train personnel to use their workstations. Common system commands can be activated by simply pointing and clicking on a menu item or icon. Thus, there is little or no need for users to learn a multitude of system commands, or to remember dozens of function key combinations. Moreover, since the commands are available for on-screen review, users can search for and activate commands that they may have forgotten.

Finally, many activities that require a fair amount of repetition and memorization on a text-based system, such as copying or moving data files from one storage device to another, become an intuitively simple task. To move files, for example, GUI users can simply select and drag a file's icon from one window to another.

Interapplication Consistency

The concept of interapplication consistency, wherein all applications designed to operate under a GUI adhere to strictly defined user-interaction guidelines, is one of the guiding principles behind the development of any GUI. By designing a system with such firm guidelines, GUI developers are able to provide users with a consistent way of doing things that extends from the desktop into every application, eliminating the possibility of presenting the user with inefficient or nonstandard means of operating the computer.

The greatest benefits of this approach are in the execution of activities that are common to virtually all applications—such as opening files, sending text to a printer, and copying or moving data. A properly designed GUI-based application will always present these functions at the same location on the menu bar.

Accordingly, a user who wants to save a file while using a word processor moves the mouse pointer to the "File" selection, then down to the "Save File" option. When using a spreadsheet or database, the command for saving a file appears under the same "File" selector, and the command is the identical "Save File" choice. When compared to the many different ways that such common activities need to be performed when using diverse text-based applications from various developers, the ease of use such interapplication consistency provides is clear.

Multitasking and Data Linking

GUIs, by the nature of their multiple window design, are well suited to the job of allowing users to run more than one application at a time. While some platforms (e.g., UNIX) already provide the ability to multitask, a GUI makes the process easier and more flexible. Not all GUIs provide the same level ofmultitasking functionality, but all GUI developers consider multitasking to be a key benefit of graphics-based computing, and are continually striving to improve this capability.

One example is "data linking," the ability to link text or graphics within one application file to related data residing in the file of a completely different application. For example, a user can link a graph in a word processing document with the spreadsheet file from which the graph's information was taken. With data linking, any future changes the user makes in the spreadsheet will be reflected in the word processor's graph.

Integrated Desktop Publishing, Presentation, and CAD Capabilities

The need for a GUI is clear for desktop publishing, presentation, and computer-aided drawing (CAD) applications due to their reliance upon multiple fonts, charting, and graphical layout capabilities. Because GUIs are not confined to displaying hardware-based text and graphic characters, they are a natural environment for the operation of these graphics-intensive programs.

Indeed, since their inception, desktop publishing, presentation, and CAD programs have had to provide some sort of proprietary GUI within the application. Obviously, a publishing, presentation, or CAD program that utilizes the same GUI as the rest of the system eliminates the need for users to learn two different interfaces. Additionally, the ability to move text, charts, or graphics from word processors, databases, or spreadsheets into a desktop publishing, presentation, or CAD applications is improved.

Multimedia

Much has been written about multimedia, the marriage of computer technology with high quality photographic images, full-motion video, and high fidelity audio. Like all electronic technologies, the cost of the components that make multimedia possible continues to drop. The popularity of multimedia related devices such as CD-ROM players is steadily increasing, and that means that some form of multimedia will play an increasingly important role in many aspects of the MIS environment.

The complexity of integrating and synchronizing the many different aspects of multimedia virtually demands that the end user be given a clear and simple means of accessing information. Graphics-based interfaces have proven unsurpassed in the ease with which they allow users to navigate through multimedia applications and manipulate graphics, animation, and sound.

Implementing GUIs in the MIS Environment

Considerations for Implementation

GUIs seem to offer a relatively painless way to improve user-to-computer interaction at a minimal, software-only cost. Also, it is true that GUIs can help to eliminate many of the end-user problems that older, text-based systems perpetuate—unintuitive training-intensive applications, arcane and difficult to remember system operation and navigation methods, and the lack of graphics creation capabilities being among the most common and problematic examples.

However, the truth is that GUIs are not the ultimate solution for every size and type of computer installation. There are numerous trade-offs involved in either fully or partially implementing GUIs into a text-oriented installation. Trading one set of administrative and operational problems for another is certainly not the goal of making changes to the MIS environment. To avoid this, the potentially hidden penalties involved in the process of moving towards a GUI-based system should be carefully considered.

System Considerations

Significantly Larger Base Platform Expense

Text-based interfaces demand far less overhead than GUIs. This is true simply because it takes less system memory and cheaper hardware components to store and display the character codes required for a text-only application. Indeed, the cost of nearly every hardware component of a system increases with the implementation of a GUI. GUIs require more expensive graphics-capable display monitors. Each GUI-equipped workstation requires more memory in order to accommodate the larger amount of data that must be manipulated in order to construct and present complex graphic images.

Additionally, the processes involved in creating these complex images are also many times more microprocessor intensive than their text-based counterparts. This requires that workstations be equipped with microprocessors that operate at higher speeds in order to deliver reasonable performance.

For many MIS installations, especially those that contain a large number of obsolete workstations and PCs, the upgrade path to a GUI-based system can make it all but impossible to protect an organization's investment in existing equipment. The reality is that many workstations and associated peripherals will either perform at marginally acceptable levels or will be totally inadequate for GUI applications.

Availability of GUI-Based Applications

One potentially serious limitation of any GUI is the lack of availability of off-the-shelf applications that are designed to take full advantage of the GUI's environment. This is not so much a problem for systems whose GUI is intrinsically integrated, such as the Macintosh. But for GUIs which are system optional, as in DOS and UNIX installations, a poorly supported GUI will bring little or no benefit to users, serving only as a graphical shell for moving between older, text-based applications.

Determining the viability of a GUI is difficult, as a system's acceptance often follows a "chicken and egg" development path. Software development companies are reluctant to design applications for a GUI until it achieves widespread acceptance in the marketplace; a GUI will languish without the support of application developers. Even the highly successful and well-supported Microsoft Windows 3.0 suffered from lackluster developer support in its earlier incarnations.

Making the right decision on which, if any, GUI to implement is further complicated by the fact that a superior design is no guarantee of success. An example of the importance of broad support is the GeoWorks Ensemble GUI for DOS-based workstations. Many in the media have declared GeoWorks the winner over Windows in both appearance and ease-of-use. In addition, GeoWorks requires considerably less system speed and memory than the current version of Windows. However, due to a lack of software support from major developers, GeoWorks Ensemble is an also-ran in the large-scale MIS arena.

Another GUI, the GEM operating system, popularized by the industry-leading Ventura Publisher desktop publishing program, is another example of the importance of having multi-developer support. Although GEM—which has been on the market since the release of the original version of Microsoft Windows—has often been heralded as superior to Windows in performance and ease-of-use, it has failed to catch on as a multi-application GUI. In fact, as a concession to the success of Windows 3.0, a version of Ventura Publisher that supports Windows has recently been released. The danger of implementing a GUI that is not well supported is clear—the lack of ready-made applications for a GUI can place an unduly heavy burden on an MIS department's programming staff.

Administrative Considerations

Increased Demands on MIS Support Personnel

The strain that is brought to bear upon an MIS organization that is implementing a GUI is not limited to increased demand on system hardware. GUIs present MIS managers with greatly increased system-level complexity. This means that virtually every aspect of GUI-based system support—including installation, training, programming, maintenance, and administration—will take more time and effort than text-based systems.

Custom Programming Considerations

An important consideration for MIS managers in determining whether or not to implement GUIs is their inherent programming complexity. This is extremely critical in the case of GUI environments that lack broad commercial developer support, as we have discussed. The code required to generate windows, icons, dialogue boxes, and other GUI elements is complex, and adds programming overhead that goes well beyond the work needed to deliver the functionality required of the application.

Pre-written code libraries for common actions, as well as the growing availability of prototyping and application generation programs for most GUI environments, can eliminate extra programming steps. These tools greatly simplify the steps required to create custom GUI programs. They allow programmers to concentrate on developing an application's functionality by automatically generating the underlying code required to make the graphical aspects of the program work.

Even the most advanced application generation program cannot guarantee that the critical aspects of programmer's GUI design will stand up under the rigors of end-user trial. Programmers must be highly attentive to the interface-related needs of end users because it is still their responsibility to ensure that the graphical aspects of the application are easy for end users to grasp. The skills required for end users to perform common activities like file creation, or navigation between hosts and servers, should be easily mastered with a minimum of instruction—even from a written guide. Equally important, users must be able to perform actions naturally, without resorting to either unusual logic patterns or ungainly input-device manipulations.

Custom programming is certainly a viable alternative for GUI application development. However, managers must not only consider the cost of purchasing GUI application development tools, but also the time, expense, and effort required in bringing programmers up to speed in using them.

GUIs and Open Systems

One of the promises heard from GUI developers is that they will help to usher in the age of truly open systems. Several industry consortiums and foundations, with members comprised of leading system and software developers, have made commitments to providing open system standards that in theory would allow installations containing hardware and software operating systems from competing companies to work interdependently without added cost or programming overhead. GUIs, most of which operate with a similar desktop/window metaphor, would seem to offer a natural environment for the development of such a cross platform connectivity scheme.

To some degree, this can be true, as in the case of the X-Windows protocol for UNIX systems, which is capable of serving as a bridge between different UNIX hardware and software platforms. However, with the proliferation of proprietary systems on the market today, and with the open system consortiums too often working at cross purposes, the reality of truly open systems in the MIS world seems remote. In this turbulent environment, GUIs may merely bring another level of incompatibility between systems, adding only more cost and complexity to the picture.

Cross-Platform Interoperative GUIs

The unfortunate reality is that many MIS installations are home to a variety of systems acquired from competing, non-compatible vendors. The previous discussions of open systems issues aside, there are still many MIS managers who are evaluating the possibility of creating consistent, cross-platform GUIs that will simplify user training. The concept of utilizing standardized, scalable software development architectures shows great promise. GUI applications developed to work on one operating system can then be easily ported to others.

Unfortunately, while scalable software development may one day be applicable to a wide variety of situations, at present there are myriad obstacles, both large and small, that make the possibility of implementing true cross-platform GUI interoperability unlikely for most mixed-platform computing environments. Even a seemingly simple task such as unifying mouse input between platforms is complicated by the fact that different GUI operating systems support mouse button input from one, two, or three buttons.

It is also important to remember that the confusion and conflicts created by the lack of a truly unified cross-platform GUI specification do not present a problem for programmers alone—end users familiar with the GUI of one platform will not necessarily be able to transfer their knowledge to the operation of a competing GUI.

User Considerations

Determining Realistic User Benefits

Much has been claimed in regards to the ways in which a GUI can transform the ways in which workers use their PCs. It is true that a GUI changes the way in which computers are manipulated by users. However, it is a mistake to assume that the mass implementation of a GUI-based system will benefit every type and level of user in an installation. The conveniences a GUI may offer will not change a user's inherent skills and abilities.

A user's attitude toward an interface is a a key determining factor in how much benefit will be derived from a GUI. For users whose computer system interactions remain for the most part the manipulation of words or numbers, the necessity of implementing GUI systems at all is brought into question. For the new user, with no preconceptions, a GUI-based system will most likely be easier to learn and use. The ultimate answers are not always easily derived by examining the problem as an either/or situation for any level of user. There are both quantifiable intellectual and more difficult-to-determine emotional factors to consider.

Potential for Decreased and False Productivity

Although GUI-based programs tend to offer more capabilities than their text-based counterparts, those capabilities may not always be necessary, nor even desirable.

In fact, there are circumstances where the very capabilities that a GUI brings to an application can make users less, rather than more, efficient in the performance of their job. No application demonstrates this problem more clearly than the current generation of GUI-based word processors.

Word processing has been one of the primary tasks performed by users since the earliest days of computing. By providing text manipulation capabilities that typewriters lacked, computer systems significantly enhanced the efficiency with which business correspondence, report generation, and other corporate writing tasks could be performed. As the capabilities of computer systems increased, so too did the benefits of computer-based word processing. Now-common features such as spell checking, automatic pagination, mail-merging, and footnote generation have truly revolutionized both the efficiency and quality of the end-user's writing.

With the advent of GUI-based word processing, this steady progression towards greater word processing capabilities took an unusual turn. Word processing application developers began to add graphical capabilities to their programs. Features such as the ability to change a document's text font size, shape, and style became common. Unfortunately, many users became overly enamored of this newfound ability to alter the look of their documents. Many GUI-equipped organizations found that, while business correspondence looked more attractive than ever before, the content of many documents was lower in quality. Even worse, many users, caught up in the minutiae of perfecting the appearance of their documents, were taking twice the time (or longer) to generate the same amount of work.

Possibly for the first time, new capabilities have in some cases detracted from the efficiency of a program. In the case of GUI-based applications, the trend toward adding more and more functions is showing no sign of lessening. Many GUI-based word processors now offer complete graphics drawing and painting capabilities, which only serve to distract the user from the text-generation functions that are all that many word processing users actually require.

GUIs can detract from user productivity in other ways as well. An important point that may at first seem humorous is the fact that GUIs make it easier for users to play arcade-style games on their workstations. One of the most often heard topics of discussion heard among Microsoft Windows 3.0 users is the excellent version of Solitaire that is included in the software package. After the smile fades at the thought of whole departments angling to "run the deck," MIS managers soon begin to realize that the potential is very large for any productivity gained from the implementation of a GUI to be lost to game-playing.

Existing Text-Based Users

In many cases, MIS managers will discover that the majority of existing end users, familiar and comfortable with the current text-based system, are lukewarm—or even hostile—to the prospect of moving to a GUI-based system.

There are numerous behavioral changes that users must make when moving from a text-based system to graphics-based interface. But quite often, these existing users, who are capable of performing all of their computer-related activities using the keyboard alone, complain mostly about the "tyranny of the mouse." For such users, who spend most of their time entering data or crunching numbers, the need to remove their hand from the keyboard in order to use the mouse can be especially frustrating. Fortunately, most GUIs are beginning offer optional keyboard shortcuts that can eliminate some of the need for mouse input. For mostly text-based applications, however, including such fundamental programs as word processors, spreadsheets and databases, the advantages of GUIs are negligible.

New and Long-Term GUI Users

Another aspect of implementing a GUI that MIS managers should consider is the very different approach to computing that end users trained solely on GUI-based systems tend to develop. New and existing end users working exclusively in a graphics-based environment can become dependent upon the GUIs many convenient mouse-based shortcuts. In the long-term, these user's analytical skills tend to suffer. They may be less knowledgeable and capable of solving problems or accessing certain data or commands without the help of the GUI. In installations where graphical and text-based interfaces co-exist, or in the case of end-users who need to utilize a non-GUI application, the potential for user confusion and the need for increased user support services exists.

Implementation Suggestions

In the final analysis, it seems inevitable that GUIs will eventually replace character-based operating systems. Still, the short term gains from fully converting an MIS installation to GUI-equipped workstations may not be cost effective. No two organizations are alike, so the only way to determine the real issues involved in the implementation of GUIs is to begin the process. If, after a careful consideration of the issues, an MIS organization decides that the time is right to implement a GUI-based system, a cautious approach is recommended.

In order to fully grasp the logistical issues involved, it is a good idea to start the implementation of a GUI-based system with only a few groups of users or departments. By proceeding in this manner, rather than attempting to implement GUIs system wide, a careful evaluation of actual productivity gains versus real costs and complexity becomes possible.

Due to the expense of converting to a GUI, even on a limited scale, it makes sense to begin implementation with the types of users who stand to gain the most. The establishment of a set of criteria, on either a per-user or per-department basis, will help define where the implementation of a GUI will be the most beneficial. Further, by dealing with the logistic and managerial problems encountered by these users (who will stretch the boundaries of the new GUI system to its limits), MIS managers will be able to more efficiently implement the GUI system for users and departments who are less demanding, and eventually, for the entire installation.

Implementation Criteria

Choosing the right users and/or departments for the initial implementation and evaluation of a GUI can be made simpler by developing a set of criteria that helps to determine which types of users will be most likely to benefit. Following are some suggested criteria.

User Demand

Users or departments who have shown an eagerness to utilize a GUI will be more committed to working with MIS to make the transition successful. They will also tend to be more creative and motivated to stretch the new system to its limits.

Increased Need for Graphics-Intensive Programs

As discussed above, users or departments that make extensive use of certain types of programs, such as desktop publishing, desktop presentation, or CAD programs, will derive maximum benefit from the implementation of a GUI.

Capabilities Otherwise Unavailable

The work of some users virtually demands a GUI. For example, any department that is considering adding multimedia production capabilities will need to implement some sort of GUI system as well.

User Turnover/Growth

Departments with large user turnover or rotation, and departments that are experiencing high growth provide an excellent medium for evaluating GUI ease of use and user training.

GUIs: Today and Tomorrow

The current state of the art in GUI environments varies depending upon the platform. The Macintosh, despite the release of System 7, remains relatively stable. The X-Windows protocol, as well as developer-specific derivatives, will continue to be a growing presence in UNIX installations. In the world of Intel 80x86-based computing, OS/2 version 2.0 shows great promise, but must compete with the highly successful Windows 3.0 and soon-to-be released 3.1.

Apple Macintosh

The GUI which is an intrinsic part of the Macintosh operating system has remained surprisingly consistent throughout the many revisions and updates that Apple has released for the system. Indeed, all the way through System 6, Apple maintained a philosophy of downward-compatible evolution rather than revolution in the design of its GUI.

With the release of System 7, the Macintosh operating system added many new features and refinements. They include full-time multitasking and object aliases, and graphic placeholders which represent a file or application that can be accessed from any convenient window/directory without the requirement that it physically reside in that window/directory's physical data storage counterpart. The best news is that the GUI of System 7 maintains the same basic look and feel of previous versions of the Macintosh system. Overall, the Macintosh features one of the most highly integrated and stable GUIs in the industry.

UNIX

Unlike other operating systems, X-Windows, the most widely used graphical environment for UNIX, is not a product but rather a set of standardized specifications that UNIX programmers can use to develop a GUI system. Technically, X-Windows is system independent, which means that an X-Window application written for one type of computer or operating system will run equally as well on another. But for the most part, X-Windows has remained within the confines of the UNIX community.

The fact that X-Windows is only a protocol rather than a complete system can make it difficult for both programmers and users. The user of one programmer's X Window interface may not find it easy to utilize that designed by another programmer. Also, programming a GUI in X-Windows is complicated and time consuming. The growing popularity of X-Window development toolkits such as Open Look from Sun Microsystems, and Motif from the Open Software Foundation, is helping to make X-Window GUIs more consistent in look and operation. The GUI building assistance that these products can provide also make it somewhat easier for programmers, but GUI programming for UNIX-based systems still remains difficult at best. Nonetheless, graphical user interfaces will be a growing presence in many UNIX installations.

Intel 80x86 Architecture

The development of GUIs for Intel 80x86 microprocessor-based computer platforms began with PC-MS-DOS, the most commonly used operating system for these platforms. However, the GUI future of 80x86 computing is moving beyond DOS. Currently, the two major competitors in the 80x86 GUI operating system derby are Microsoft's Windows 3.0 and IBM's OS/2 with Presentation Manger.

Windows

As discussed earlier in this report, Windows is the GUI of choice for most 80x86 PC users. The current Windows, version 3.0, and the soon to be released 3.1, which is an evolutionary update that will add various performance improvements, operate as an adjunct to DOS, although certain activities, such as screen and keyboard input/output bypass DOS. Windows, when running in Enhanced mode, also bypasses DOS's 640K memory addressing limitation.

While there is no doubt that Windows has achieved wide popularity and is being used on millions of PCs, that does not mean that the current version is without serious drawbacks. The complaint voiced most often is that Windows is simply too slow. It is slow both in updating the graphically-arrayed display and in the time it takes to perform various processor and disk-intensive activities. For many Windows 3.0 users, far too much time is spent watching the Windows hourglass icon.

This is especially true in the case of Windows 3.0 used on 286 and slower 386 and 386sx processor-based workstations. The problems caused by slow operation are not fully measurable in the amount of time it takes to perform a given task, but also in the level of frustration it causes users as they wait for tasks to be completed. The proliferation of high-speed workstations is helping to eliminate this problem, but the cost of these more expensive systems should be considered when choosing a GUI for a large-scale installation.

While full development tools are available for Windows 3.0, they are unpopular with some application programmers, who find them to be a somewhat unwieldy development environment. Nevertheless, nearly all the major software developers, including WordPerfect, Lotus Development, and Borland International, either have delivered or are developing Windows-specific versions of their applications. Of course, Microsoft has made available Windows-specific versions of its popular software packages, such as Microsoft Word and Excel.

Microsoft is also planning a 1992 release of a system-independent versions of Windows, called Microsoft Windows NT (New Technology). Windows NT breaks free from the constraints of DOS to become a standalone operating system. Microsoft's plans call for Window NT to be usable on a variety of hardware platforms by providing a downward-compatible (to DOS-related Windows 3.0), scalable architecture.

OS/2 with Presentation Manager

IBM's OS/2, with Presentation Manager (its GUI component) is also a replacement rather than an enhancement to DOS. OS/2 lags behind Windows in developer support by some distance, but is not yet dead and buried, as many have claimed. The soon to be released Version 2.0 offers many improvements in speed and flexibility. Its application development environment has received praise from many in the software engineering arena. One of the biggest complaints heard about OS/2 has been its hunger for system memory, but this new version brings OS/2's requirements in line with those of Microsoft Windows 3.0.

80x86 GUIs: Future Developments

Both IBM, with OS/2 2.0, and Microsoft, with Windows NT, have announced plans to implement the capability to run applications designed for the competing operating system. Beta versions of OS/2 2.0 feature the ability to run Windows for DOS applications. Windows NT promises an OS/2-Presentation Manager add-in at a future date sometime after the product's scheduled 1992 launch. How capable either system will be of running them at an acceptable speed and without other operational problems remains to be seen.

Since both OS/2 2.0 and Windows NT are in the infancy of their life cycle, no hard analysis will be possible for quite some time. But, regardless of promised capabilities or features, or the machinations of the respective developer's marketing efforts, it is the performance and capabilities of these systems that will decide which, if either, becomes predominant. Microsoft has a substantial lead in installed base, but many of the nation's large MIS organizations have not yet decided which upon which GUI to standardize. Many have decided to take a wait-and-see approach before making any sort of massive GUI investment.

GUIs: Tomorrow and Beyond

GUIs will continue to evolve and become both more intuitive and more efficient. Like many key developments in the evolution of computing, GUIs will not truly transform how people use computers; rather, GUIs will bring computing more productively into the performance of everyday work activities.

Improvements in Technology

Like everything else in the computer industry, GUI-based operating systems will continue to improve, as will the hardware platforms that they are designed to operate on.

More Intuitive Interfaces

New products that show the path developers and manufacturers are taking to move graphical user interfaces ever closer to the ideal of providing users with a totally natural and intuitive means of interacting with a computerized device, are just now coming into the marketplace. One of the most interesting is the new breed of notepad computers, which let users operate the computer in much the same way as they would write on a sheet of paper. These computers, which are about the same size as a regular paper notepad, make the GUI an intrinsic part of the operating system.

Using an attached ballpoint pen-like stylus, users can interact directly with the GUI's workplace surface. The stylus operates in two ways. It can act as a pointing device, much like a mouse; and it can be used to write, like a regular pen or pencil, eliminating the need for a keyboard in many situations. At this time, competing systems are being developed by Microsoft, whose notepad GUI is called Pen-Windows, and Go Corp., whose GUI is called PenPoint. A third notepad GUI is offered with Momenta Corp.'s Momenta notepad computer. Which, if any, of these GUIs becomes the standard for notepad-based computing will not be clear for some time to come.

Voice Recognition to Virtual Realities

Beyond the evolution of GUIs into systems which aim to improve traditional eye/hand interaction between users and their computers, like the notepad computers discussed above, lie several technologies that promise to go beyond—providing purely intuitive verbal and/or physical control of computer systems. Although it may be many years before such systems are widely available, they provide a glimpse of the ways in which we will all be able to use computers in the future.

Voice recognition is one technology that is moving users away from the keyboard. While the technology to record speech, and even to have computers "learn" a limited number of spoken commands, has been available for many years, only recently has technology begun to approach the capability of producing systems that can understand the spoken word and carry out complex commands with acceptable reliability. In 1991, both Apple Macintosh and Tandy PC versions were released that featured audio input and storage capabilities built into the base system, evidence of the growing importance sound plays in computer operation. The possibility of developing fully functional PC systems that require neither graphics display nor keyboard input is becoming a reality.

At the opposite end of the GUI spectrum is the experimental work now being done in the development of "virtual realities," computer systems that take the concept of a GUI to its extreme by creating an artificial environment in which users can interact with the computer by making natural movements. In a virtual reality system users can control events simply by walking, touching and gesturing.■

This report was developed exclusively for Datapro by Robert Levin. Mr. Levin is a principle of TECH-WRITE/MED-WRITE, a technical and medical writing and marketing support organization located in Newtown, PA. Mr. Levin has written extensively on many aspects of computers, data communications, and high technology. He also consults and speaks on the development of systems and processes that optimize interpersonal and human-to-computer interactions.

Workstations: Current Market and Future Trends

In this report:

Datapro Summary

When they were initially introduced, workstations were positioned as 16-bit-based, electronic computer-aided design (ECAD) systems. Now, however, today's workstations contain more power than most of the mainframes available in the mid-1980s. With 32-bit processing power available in today's workstations, they are well-suited for business, financial, and industrial markets, as well as their traditional engineering niche. The potential for workstations of the future seems limitless.

Introduction

When workstations were introduced in the early 1980s, the distinction between them and personal computers was easier to understand. Workstations cost more, performed advanced functions, and had powerful graphics and floating-point capability unavailable on personal computers. Today entry workstations are less expensive than some personal computers but still have more powerful graphics and floating-point capabilities and depend on the UNIX operating system.

Workstations were originally introduced as electronic computer-aided design (ECAD) devices using 16-bit architectures, but workstations rapidly expanded to 32-bit processors. Workstations passed through two stages of development and are currently in their third stage, as shown in Figure 1. Early operating system support was vendor specific but has shifted to UNIX. By the third stage the market expanded to include financial, business, and industrial users.

This Datapro report is a reprint of Chapter 10, "Workstations," pp. 173-195, and Chapter 14, "Challenges of the Fourth Wave," pp. 251-267, from *Professional Workstations* by Thomas F. Wheeler. Copyright © 1991 by McGraw-Hill, Inc. Reprinted with permission.

This shift in the market can be seen in the 1987 approximation of the workstation markets (shown in Figure 2). Over 60% of the market was focused on engineering disciplines, and software development comprised another 20%. Some financial companies were beginning to use workstations. By 1990 the market had spread well beyond the technical community as prices equaled or bettered those of personal computers and easier interfaces were introduced for UNIX.

Applications consumed workstation power as fast as it could be provided. The most sophisticated graphics applications are those that produce images and animation. Mission-critical applications can also benefit from the performance available in advanced workstations. Since prices have fallen as performance has increased, powerful processors with high-speed graphics capabilities can be used for even simple applications. This power benefits personal and group productivity when it is correctly applied. Further commercial use is expected as productivity applications become available. Productivity studies have demonstrated that the payback period is short when these applications have been tailored to business' needs.

Projections show a continual improvement in workstation performance through the rest of the century. Business estimates

Figure 1.
Workstation Development Stages

STAGE 1	STAGE 2	STAGE 3
16-BIT MICROS 16-BIT BUS CISC VENDOR OS DISPLAY ENGINEERING/ SCIENTIFIC APPLICATIONS HACKER INTERFACES APOLLO DOMINANT	32-BIT MICROS 32-BIT BUS RISC INTRO UNIX STANDARDS EXPANDED APPLICATIONS SOME LOOK & FEEL SUN/ HP/ DEC APOLLO LANS	32-BIT MICROS SOME 64-BIT 32-BIT BUS RISC UNIX STANDARDS MULTIMEDIA NEW APPLICATION AREAS LANS GROUPWARE SUN/HP/DEC APOLLO/IBM/DG
1981-1984	1984-1988	1988-PRESENT

show a shrinkage in revenue derived from hosts and minicomputers for the same period, while workstations command a higher portion of the total computing dollar.

New technologies provide opportunities for designers. As developments in peripherals continues, characteristics of desktop computing will be modified by new applications. Each new design will challenge the conventional system's wisdom and place an additional strain on systems integration.

Imaginative approaches vary within each vendor, but a general trend to define product with sufficient granularity to work on desktops or servers is developing. At the high end of the configuration spectrum are powerful network servers, which combine large disk capacity with high performance. Servers will expand their computational function and their ability to handle several clients or workstations. The level just below the network server is the department server, which has a smaller capacity.

Vendors can now purchase the parts for assembling workstations with little proprietary design required. RISC chips that can be obtained from merchants, coupled with widely available support boards, establish an environment in which new companies can enter the market easily. The relatively low cost of producing standard software has aided the process. Everything from processor chips to application packages can be purchased by a vendor. A workstation can be made available to the market in as fast as 6 months.

Operating Systems

As the power of the workstation expanded, the requirements for desktop operating systems changed from those defined for DOS. Functions such as multitasking, networking, and improved usability became important for successful operating systems. Multiuser support became a requirement for servers. Multitasking provides the capability of running several applications simultaneously. Networking connects the workstation to the LAN, and improved usability provides simpler access to applications and control systems.

UNIX—The Operating System of Choice

Competition for the operating system that will be chosen continues to be fierce, since the operating system will determine the hardware market share. OS/2 was a contender for high-end personal computer software but was limited by its 16-bit design. The system is targeted for other workstations, but the delay in the appearance of the 32-bit version beyond 1990 has caused the system to lose much of its initial appeal. Both the delay and the design limitations delayed the acceptance of critical hardware, provided additional growth for DOS, and confirmed UNIX as the preferred workstation operating system.

The name UNIX was derived from MULTICS, which was developed at MIT in the 1960s in joint work funded by AT&T and General Electric. UNIX was developed at Bell Laboratories by Kernighan and Ritchie in 1972.[1] Since AT&T was restricted from developing computers because of their 1956 consent agreement, the system was used early at AT&T locations and within the academic community. Its introduction in universities resulted in even wider popularity as graduating students carried their enthusiasm into industries. The simple and open design contributed to UNIX's early popularity as universities tailored the code and created many interesting programs using the system. The defined system layers isolated functions in the operating system and made it possible to port the system to other hardware.

Eventually outside implementors gained access to the system and provided external versions. Initially interest was focused in engineering and scientific groups, but gradually it extended to the commercial community. The ability to capitalize on a rich base of applications has made the system very popular with workstation implementors as they ported the code to their hardware architectures.

This portability is a good reason for UNIX's popularity. Three characteristics contribute to the ease of porting. The principal elements of the system are coded in a high-level language C, which enables recompilation for different instruction sets. Since the system is composed of primitives, it is possible to modify them for different operating environments without a great deal of difficulty. Consistent interfaces are used throughout the system, so implementors could use the same commands on different processors to access system functions, files, or the drivers used for peripheral attachments.

An Explanation of UNIX Terms

A user's introduction to the system begins with data support that has a tree-structured hierarchy based on directories and files. Data files are handled as linear byte representations, which enable application programs to identify

Figure 2.
Workstation 1987 Markets

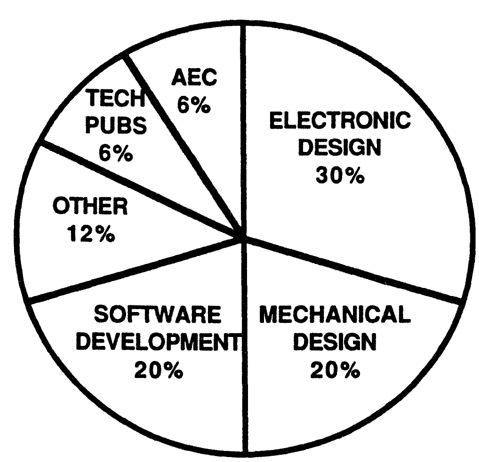
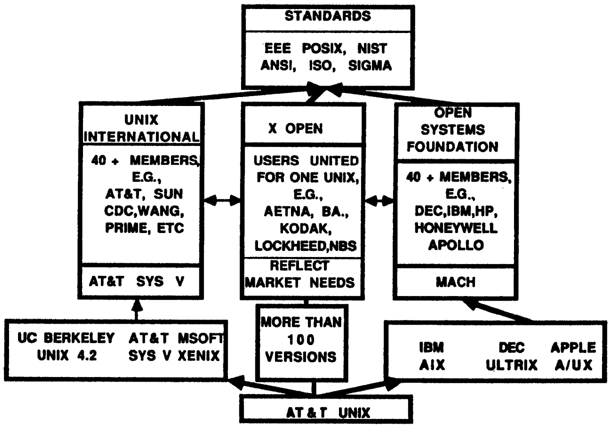


Figure 3.
UNIX Interactions



and recognize parameters that optimize their performance. Peripheral devices can be treated as simple extensions of the file system, which simplifies their access through programming. Programs become sensitive to the content of the byte representation, but the file structure is identical for all system programs. This approach is different from traditional techniques that define files as records and fields. The file system has proven versatile for scientific processing and is also gaining popularity in commercial enterprises.

In UNIX all programs become forms of executable files. Another design element is *processes*, which are execution instances of programs and are comparable to tasks in other operating systems. Multiple processes can be running simultaneously in supported systems. *Forks* are calls that initiate new processes within the system. Interprocess communications occur with *pipes*, which can be considered conduits between different processes.

At the heart of the implementations of UNIX is the *kernel* that interfaces to the hardware and provides services for other software in the system. The kernel schedules processes, allowing their paced execution. Interprocess communications and device handling is also handled through the kernel. The kernel provides the necessary resources, such as storage allocation, for system functions and is the transparent foundation for the multiuser and multitasking support.

The *shell* or command interpreter defines the interfaces for other functions within UNIX. The shell communicates with the kernel to request actions or data from the system. Although shells normally execute commands sequentially, they also permit the user to define asynchronous commands, which avoids wait periods. Since the shell is not part of the kernel, the shell is easy to modify, and, as a result, a number of different shells have been implemented that have wide acceptance by users.

A number of identified weak points have been corrected by recent versions of the system. Although this is true, some features remain weaker than vendor alternatives and contributes to concern about the level of security. Many vendors have provided versions of the operating system that conform to the toughest government standards.

The poor usability for unskilled users has been another negative factor for users who prefer the simplicity found on some personal computers. Vendors and user groups are working to address these problems, but this effort will take time. UNIX is still not as simple as most personal computer systems, but companies have provided capabilities that mask its complexity. Apple, for instance, provides its highly regarded Macintosh interface for UNIX. Another example is SCO's introduction of Open Desktop for UNIX, which simplifies the end-user interfaces.

The Appeal of UNIX

An incentive to write applications on UNIX is its early support for the C programming language.[2] The language was developed by the designers of UNIX, and it became the choice for high-performance applications and the natural choice for code migration between operating systems. Advanced tools such as Programmer WorkBench (PWB) have also made UNIX attractive.

Network support is available through standard TCP/IP protocols. One scenario for personal computers is to connect them on Ethernet to UNIX applications. The personal computers can continue to run on their preferred operating system and communicate directly with the UNIX applications. Using this approach, companies can increase their capability while retaining their existing computers and applications. Increased function is integrated in the network, enabling programmers to focus on emerging technology.

Although AT&T dominated the early UNIX design, many groups improved the system. The Department of Defense commissioned the University of California at Berkeley to rewrite the system in 1980. This work produced Berkeley Software Distribution (BSD), which is widely used.

Code portability was an early reason for acceptance of UNIX, but as seen in Figure 3, the UNIX world was complex by 1988. From the AT&T introduction over 100 different versions were produced by vendors. Radically different approaches were adopted by IBM and AT&T, which resulted in two separate camps.

To bring order to the increasing chaos, a number of approaches have been made to standardize specific UNIX definitions. UNIX International was established by AT&T and Sun Microsystems to define a standard based on AT&T's System V. Extensive work was performed to broaden the coverage, and eventually over 40 companies joined this group and worked toward a common set of definitions. The group has produced definitions and an implementation of the standard that has been widely adopted.

Open-Systems Foundation (OSF) was established by IBM, DEC, HP, and others in May 1988 to define a standard based on work done outside of AT&T. Eventually over 40 members also joined this group, which by 1989 had adopted Carnegie Mellon's Mach as the kernel.

X/OPEN Consortium Ltd. was founded in Europe to adopt a set of standards for UNIX-using companies. The consortium focused on application portability for the different versions of UNIX. The extent of the consortium's interest includes the definition of a complete open-systems protocol that includes languages, database, graphics-user interfaces, and networking protocols. X/OPEN adopted the IEEE POSIX definition for UNIX and issued a portability guide with test cases that demonstrated conformity.

UNIX will become more popular as shrink-wrapped applications become available, i.e., the chaotic mixture of different implementations will be replaced with standards. Worldwide standards groups are focusing on a complete definition. They have defined a version of the system known as POSIX (IEEE #1003.1). NIST, formerly the National Bureau of Standards, leads the United States government efforts. ISO is defining networking standards for inclusion in UNIX. SIGMA, a Japanese government group, is defining hardware and development tool standards including UNIX, and the Department of Defense remains involved through DARPA, which funded the Carnegie Mellon's restructuring of UNIX.

New standards have expanded the functions within UNIX. Windowing, graphics-user interfaces, and communications protocols have been defined. As an example, the X Windows system was introduced through the combined work of DEC, IBM, and MIT in project Athena. The X Windows system provides common interfaces for intelligent workstations and supporting terminals. Another example is Motif, which has been adopted as a user interface running on X Windows. The three-dimensional, personal computer-like appearance provides a common interface across multiple platforms. Internal definitions permit varieties of logical push buttons to handle different states. Popup windows provide support for bulletin boards as well as normal processing. The Motif User Toolkit provides a graphical interface layer based on X Windows internals. Included in the internals are definitions of widgets that define text and graphical storage space, push buttons, and sliders. Containers have been defined for list of strings as well as collections of special texts and lists. Popup aids are in the tool kit to facilitate screen creation. These tool kits make it possible for application developers to define their own windows.

Although there are many positive features about UNIX, there are also some drawbacks. Complexity has been a hallmark of the early versions, making it necessary to arrange for extensive training before using the system. Trained personnel remain scarce, which represents a problem for UNIX-using companies. Different versions and levels affect the movement of software between machines, but even in the worst case, portability is achieved in less time with UNIX than with other operating systems. Availability of general, low-cost software remains limited to certain application areas. Shrink-wrapped software is only now beginning to emerge.

Once UNIX was introduced to the commercial market, the number of installations has consistently multiplied and users include premier scientific and commercial companies. Realization of the system's positive features led to acceptance in the business community. UNIX's continued acceptance within the professional and commercial communities seems guaranteed. In addition to being portable, the system is scalable to computers of any size, from workstations to supercomputers. Since portability permits vendors to move established applications to their hardware architecture, most workstation announcements provide UNIX at least as an alternative. UNIX is an integral part of open-systems approaches, so it is being adopted by larger computer vendors also. This migration to larger and smaller computers ensures its continued success through this century.

Leading Vendors

By 1989 the market for workstations had become the fastest growing segment of the computer business. Continued advances in technology resulted in a variety of alternative products. Many different price and performance alternatives are available for desktop, deskside, and server units.

Four companies have been the leaders in this market since 1986. These leaders are Apollo Computers, Sun Microsystems, HP, and DEC. HP acquired Apollo in 1989, bringing it briefly to the number one position, but Sun Microsystems regained its leadership by the end of 1989. In the third place was DEC, which had significant gains, especially in Europe. Intergraph and Silicon Graphics are the next largest vendors.

Since this market responds rapidly to technology leadership, changes will occur as different companies introduce leading-edge support. Areas such as superworkstations remain niche markets with limited customers, but these machines are indicative of characteristics of tomorrow's low-cost products. We examine the three market leaders as examples of how workstations have been evolving.

Sun Microsystems

The market leader for several years has been Sun Microsystems, based in Mountainview, Calif. Sun (Stanford University network terminal) Microsystems adopted simple business objectives when it was incorporated in 1982. From its earliest machines, its goal was leadership in price and performance of professional workstations. Quality became a hallmark, which coupled with its capability to introduce workstations rapidly, contributed to its success.

Sun's west coast entrepreneurial approach established an attitude toward competition that presented a refreshing openness to workstation customers. Sun's early products, shipped in 1982, stressed open-systems design. Since its founding, the company has shown continual growth with steadily increasing revenues. The emphasis on open design and standards has special focus in the areas shown in Figure 4. Sun's leadership in UNIX is well known, but the company also advocated other industry standards. This standard approach gained the support of CAD/CAM vendors and their customers.

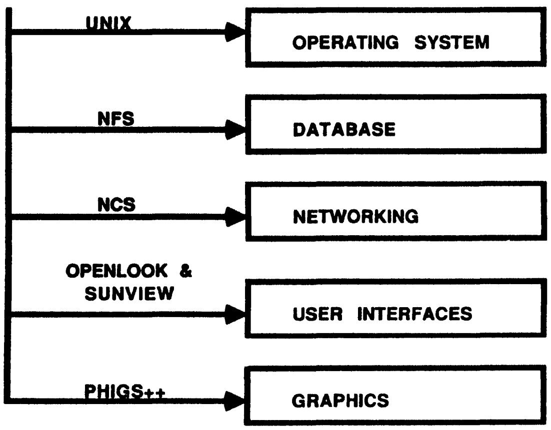
By 1990 Sun was still using three chip architectures for its workstations and servers, as shown in Figure 5, but Sun was moving to replace all of its machines regardless of price and performance with its RISC designs. Sun had relied on industry standards to bridge the different processors and provide continuity for application developers.

Sun's original workstations were based on Motorola M680X0 processors and support chips. These systems sold on the strength of their engineering and software that could run on UNIX. Since the company stressed open systems, Sun applied early and effective pressure on the industry leader Apollo, eventually unseating it. By 1989 the company was replacing the Motorola units with SPARC alternatives.

The company's IBM PC-compatible computer, the Sun 386i, combined UNIX and DOS operating system capability on Intel 80X86 CPUs. Although the revenue from this line was sizable in 1989, this product line was also being replaced by lower priced SPARC alternatives. Sun supports coprocessing on other workstations to enable the running of DOS software. Called SunIPC (Sun integrated personal computer coprocessor) this function supports personal computer applications on Motorola- and RISC-based workstations. Coprocessing permits the workstation user to run both types of applications.

Sun's initial application packages were for electronic engineering, but mechanical engineering followed shortly. In its early years Sun broadened its product lines into other

Figure 4.
Sun's Focus on Standards



engineering and scientific applications, and more recently Sun expanded its market into the commercial and financial services market.

Sun introduced a family of RISC workstations and servers based on the 32-bit SPARC with accelerated floating-point support. Originally these features were available only for higher priced, better performance units, but eventually they expanded the product offerings downward in price to include models priced to compete with personal computers. Advanced SPARC chips, which are projected to handle around 100 million instructions per second, promise even higher performance.

SPARC gained wide acceptance through the company's open-design approach. This resulted in several technological offerings and further expanded the use of SPARC technology in other workstations. A consortium established to propagate the architecture, SPARC International, encouraged multiple implementations and many competitive products that could use Sun's software.

SPARC was originally introduced in workstations. This line was broadened to cover other faster workstations and servers, including the SPARCStations. With the introduction of SPARCStation 1 in 1989 inexpensive, but high performance, units were available that used gate array technology. The flat-profile processor uses powerful processor chips and dual disk drives. Using fewer chips than personal computers, the units consume less energy than a 100-watt light bulb. The flat appearance is ergonomically appealing. It is housed in a low-profile cabinet, with powered processing and a large memory. Two high-megabyte disks can be included in the 2.5-inch cabinet. In addition to an industry standard SCSI attachment, the design features Sun's unique bus.

Sun has continued to introduce higher performance SPARC workstations while introducing lower cost processors that were priced below entry personal computers. The SLC incorporates the mother board directly into the display chassis. Continued reduction in sizes has permitted the company to develop the full support on the board.

Different levels of graphics support is available for users of each level of processor performance. As an example, the graphics processor (GX) provides image processing with over 500,000 two-dimensional images painted per second, and a comparable unit provides the same function for the SPARCStation. Color support is available on all processors.

SunOS (Sun operating system) is Sun's UNIX operating system that spans the company's hardware architectures. Cross compilers enable systems to produce code for other workstations. The operating system is the base of Sun's open-systems efforts.

Sun's entry in graphical-user interface is called Open Look. Based on X View this window-based icon system provides a simple approach to program development and interprogram communications. A migration path has been defined from earlier SunView packages to this more powerful user interface. This migration is provided through the use of a toolset that has support for the new design. To simplify icon usage, a package of routines called Desk Set is provided to users. Open Look is available to other vendors in whole or individual parts and is an important contender for industry standards.

Communications linkage is provided through a family of communications products known as SunLink. Products support LANs as well as connections to other vendor protocols. Local- and wide-area connections use industry standard protocols.

Both workstations and servers retain an open-system design with support for standard languages. The workstations support Fortran and C but also include support for Pascal, Modula 2 and C++. COBOL is available through third-party companies, often provided with target-testing environments. Server software includes a Database Excelerator that enhances performance for database vendors implementing their solutions on SunOS.

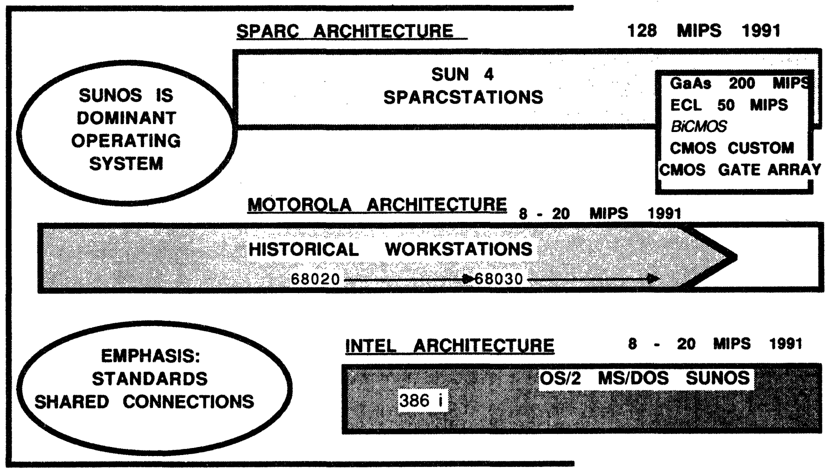
Sun has contributed designs that have been adopted by other vendors. Sun supports heterogeneous LAN through its open network computing (ONC), which is a modular set of protocols and services. It addresses three major support areas in networked environments that include resource access, application design, and network management. Data exchange across many machines is supported though the network file system (NFS). NFS was designed to be portable on several operating systems.

NFS is widely used for client/server implementations. Clear interfaces provide implementations with direct access to the server. An example of the clarity of interface is the mount command. The mount command activates a directory linkage between a client and server. A client may issue many mount commands to either one or several servers, establishing multiple connections. An unmount command disengages the connections.

The remote procedure call (RPC) enables programmers to access services across a distributed network, using an accepted application programmer interface (API). The command makes a number of library routines available to the programmer through which the user can access the most appropriate computer to perform a particular job, such as using a supercomputer for computational-intensive processes. Each procedure appears to be local to the programmers that use them. External data representation (XDR) masks the differences between data storage formats for client/server pairs. Transport-independent RPC can use a variety of network-transport protocols. Naming services provide an enhanced capability to update multiple servers on the Sun network.

ONC has been widely adopted by the personal computer and workstation world. It is implemented by major workstation vendors and is available on personal computers, including IBM's PS/2 models. Over 250 licenses were in place by the end of 1989, and more are expected as the process continues.

Figure 5.
Sun Microsystems' Architectures



In 1989 the company experienced product and operations problems connected with their internal processing system. Sun recovered from these problems to achieve even a stronger position in the market. Sun's approach to openness has resulted in skeptics questioning whether it could survive while sharing designs so openly. Although the evidence is still accumulating, Sun appears to have produced a powerful family of workstations and software that have made it the industry's leader.

Hewlett-Packard

Hewlett-Packard was founded by a couple of electrical engineering graduates from Stanford University on January 1, 1939. David Packard and William Hewlett designed the company's first product, which was an audiooscillator used by Disney studios for the movie *Fantasia.* The company was incorporated in 1947.

HP now develops over 10,000 products, including midrange computers and workstations. Its first computer was announced in 1966 and was targeted for gathering information from instruments. Both HP's computer and peripheral businesses have been successful. HP's laser printer family has been widely accepted by hardware from all vendors, and peripherals by 1989 represented 27% of its total business.

Each company creates a culture that makes it unique in the business world. HP is noted by its customers and employees for its professional straight-speaking attitude. Its products are known for quality and the excellent service that backs up the quality. The company devotes 10% of its revenue to research and development to pursue other products and perfect those lines already in existence.

HP has been a major participant in the workstation business for several years with a range of products reaching from Intel-based Vectra personal computers on the low end to very high speed workstations on the high end. Although each product line retains its autonomy, the personal computers and workstations have been coupled under industry standard protocols.

To augment its already strong role, in 1989 HP acquired Apollo Computer Company. Although the acquisition propelled the company into the number one spot in workstation sales, it also resulted in the typical confusion that follows such a merger. The merger resulted in five distinct workstation architectures as shown in Figure 6.

Just as each company has a culture, acquisitions and mergers often force a collision of cultures. Apollo brought the culture of an eastern entrepreneurial company to an established western company. The combination of the two workstation lines caused an upheaval between the cultures.

In addition to those workstations acquired from Apollo, HP provides a full line of workstations and servers, which begins with personal computers using Intel 80X86 chips. The personal computers, known as Vectra, were introduced in 1984. Using standard operating systems, OS/2 or DOS, HP can introduce more powerful processors when new chips are available from Intel. The company assumed a leadership role in the early drive to EISA bus support in 1989.

Building on its expertise, HP introduced its first desktop mainframe in 1982. This generation used Motorola 680X0 chips and the UNIX operating system as well as native systems. HP also produces products using high-speed RISC processors that are based on HP design. HP/UX is HP's UNIX package that supports its product line, using the POSIX standard with Motif for the presentation services. In keeping with the company's use of standards, HP use X Windows in the user interfaces. In addition to UNIX and DOS, the company supports two other internal operating systems.

HP has defined a number of special support packages that have been adopted by other vendors. NewWave is a user environment that runs on Microsoft Windows and DOS, providing ease of use for desktop computer users. The package uses icons with pick-and-click capability to access information. Based on an object design structure, the software combines textual, numerical, and graphical information to provide a combined interface for the user. Networking support extends these interfaces to connected processors. As in the case of other ease-of-use packages, there is a rich use of icons, objects, and task management routines. Companies such as AT&T, Canon, Data General, and NCR use NewWave for their office automation packages.

Another significant tool is CASEdge, which provides full-cycle coverage for software development. Using a Motif interface, the tool provides access to different programming packages. The first of these is a program editor, builder, debugger, and development manager called the Encapsulator which facilitates porting the tools to an HP/UX base.

The Apollo purchase by HP resulted in a combined company with five different workstation architectures that overlap in many areas of support. Evidence of convergence in the product line was a joint announcement in 1990 of

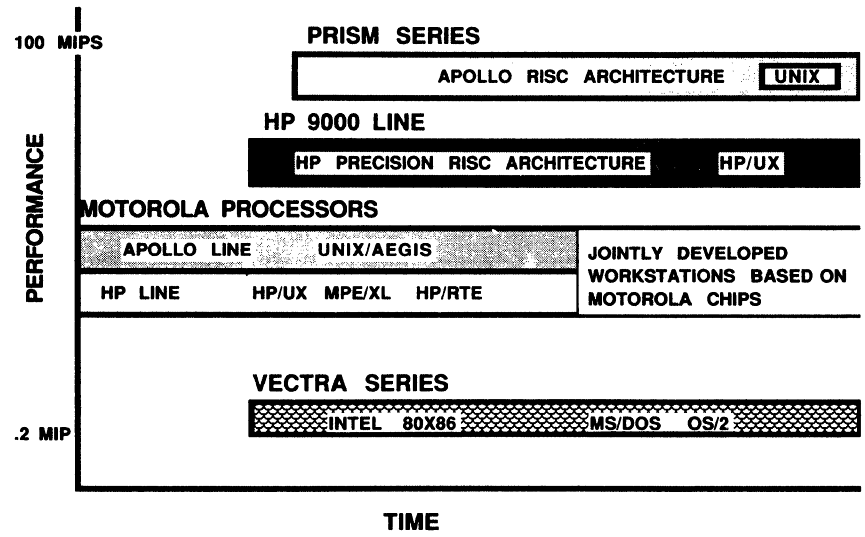


Figure 6.
Hewlett-Packard Families of Workstations

combined desktop units based on Motorola chip architecture. The combined processor can support programs using either DOMAIN or HP/UX operating systems.

HP will continue to be a strong participant in the workstation environment, since the company is well situated in the science and engineering community with powerful processors and support software. A modification of strategy will ensure effective participation in the broad-based market. These changes include a stronger focus on the needs of large commercial enterprises.

Although HP purchased Apollo, it is worth examining this workstation pioneer separately. Apollo was founded in 1980 by seven entrepreneurs from the Boston area. The company's professional workstations were initially based on Motorola processor chips. At its introduction the 250 KIPS engine provided significant desktop computing for $45,000. The machine's emphasis on floating point and graphical performance established a niche market that expanded to the workstation business.

Apollo

The early workstations used proprietary designs but the company shifted to open design in response to competition. Figure 7 shows the progression of architectures within Apollo. Its workstations used two distinct hardware architectures. The first was the Motorola architecture, with products developed on M680X0 chips. There were three levels of price granularity that offered a wide range of alternatives. At the very high end, the company formed a joint offering with Alliant Computers for a high-performance server.

Reflecting the trend toward using RISC architectures, Apollo introduced PRISM (parallel reduced instruction set multiprocessor) to achieve even higher performance on its fastest workstations. Initial processors reached a performance of 44 MIPS and 12 MFLOPS (million floating operations per second) on a 64-bit processor in 1990 or roughly 60 times that of the DEC VAX. By packaging processors in parallel, computational levels of above 160 MIPs were achieved. These processors could be used as servers or high-speed workstations. PRISM processors utilize the power of the RISC chips to perform graphics functions.

Apollo's system design philosophy is based on distributing power to individual workspaces and connecting them using the DOMAIN network. Their DOMAIN operating system combined part of UNIX V.3 and Berkeley BSD 4.3 as well as support for Apollo's older Aegis operating system. The system provides connection mechanisms for local work groups through network management.

The DOMAIN token ring support is an important industry, interconnection architecture. It performs well for local connections but requires additional translation software to connect to broader nets. An integrated PC/AT bus supports the DOMAIN/PCC connection to personal computers, while Ethernet connects larger workstations. Using DOMAIN, personal computer programs can run in one window with concurrent UNIX tasks in another.

Apollo scored a significant first by charging less than $5,000 for their low-end Motorola workstations. It was the first workstation that cost less than a personal computer. Between 1983 and 1988 Apollo saw its performance move from relatively slow processors to exceptionally powerful workstations. The applications using workstations also expanded from simple electronic design to extensive numerical models that once required the power of supercomputers.

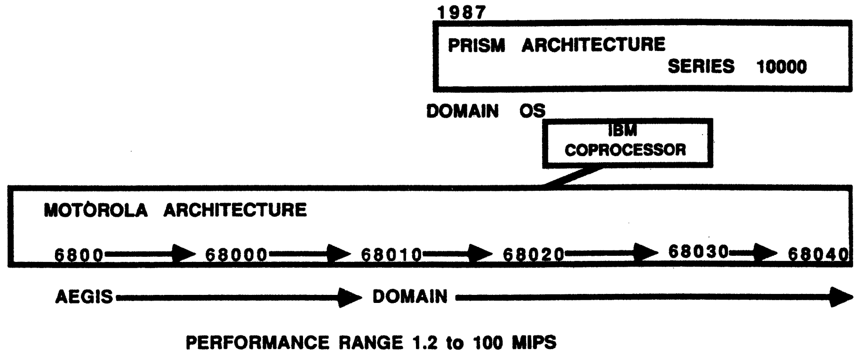
Apollo's network-computing system (NCS) is a set of utilities, based on its network computing architecture, which splits tasks between several processing units. The support provides transparent connections to other computers on the network. Access to network data is through a RPC or network interface definition language (NIDL). Additional utilities are provided to facilitate the use of NCS. An example of these utilities is the network backup utility that provides backup for network-connected workstations. NCS has been adopted as a distributed data technique by many vendors, including IBM and DEC.

Apollo has pioneered many innovative approaches that have been adopted by the industry. This capability, when combined with HP's technology, should ensure their continuing influence on future workstations.

Digital Equipment Corporation

DEC was founded in 1957 by Ken Olsen and Harlan Anderson in an old textile mill in Maynard, Mass. DEC's success has made it the second largest computer company

Figure 7.
Progression of Architecture Within Apollo



in the United States and guaranteed its participation in most product areas. When it was founded, the company reflected the entrepreneurship that led to the development of many of the companies around Route 128 near Boston. DEC's initial success was in minicomputers, and its early workstations reflected its heritage.

The company introduced its first 32-bit workstations in 1984 with a series of VAXStations. These used DEC's prevalent minicomputer architecture that was introduced in 1978. These workstations supported the virtual management system (VMS) as the operating system and its support products. DEC benefited from a technical community familiar with VAX applications. These experienced customers could easily move software to the new workstations. VAX became a major performance benchmark for testing competitive engineering and scientific products. The use of the same architecture for workstations appealed to existing engineering and scientific shops, since they had a repertoire of VMS software.

In 1989 DEC shifted from its historical VAX-only support to adopt the three-pronged approach shown in Figure 8. This strategy is based on continuing older architectures while introducing RISC processors with new software solutions. The three-fold strategy continues the VAX line with a series of workstations called VAXStations. These processors enhance the commitment to established DEC customers and uses programs tailored from the original minicomputer series.

DECStations, announced in 1989, provide high-performance computing for desktop computers through the use of RISC chips from MIPS computers of Sunnyvale, Calif. At the high end DEC provides DECSystems, which use the architecture for its servers. Using the same family of chips, workstations were introduced whose desktop units performed well using the UNIX operating system.

At the lower level of its strategy, DEC affirmed its interest in personal computers through a joint deal with Tandy and developed connectivity programs for Apple's Macintosh computers. The development of network software enabled these processors to participate actively in the DEC network.

DEC's VMS is a vendor-specific system, which was designed as a multiuser system to support virtual memory and extended functions. VMS received much support because of its virtual memory; event-driven, interprocess communications; priority scheduling techniques; and large programs with easy-to-use interfaces. Since VMS can be used on the range of machines from the VAXSystems to the large-scale clusters, the operating system represents a DEC-portability vehicle.

Since VMS had not penetrated many segments of the scientific community, DEC introduced UNIX for its VAX computers. For a long period users have questioned whether ULTRIX (DEC's UNIX) received adequate support from the company. The introduction of DECStations placed further requirements on ULTRIX.

DEC participated in the development of X Windows at MIT, and released an extended version called DECWindows. The company's adoption of the standard has contributed to the acceptance of X Windows. NFS, defined by Sun Microsystems, was included in the system for connecting network files.

DEC supports a set of languages on both UNIX and VAX/VMS operating systems, including standard languages such as C++, Cobol, Fortran, Pascal, Lisp, PL/I, and Modula 2. These provide users with a source of application software for either system. UNIX application developers are providing software for DEC's offerings since these computers have become a major participant in the marketplace. The company stimulates new applications by working jointly with users who have designed applications using DEC equipment.

VAXStations represent both a continuity and discontinuity. With their introduction DEC confirmed its commitment to the prevalent VAX architecture but provided performance characteristics that were different from previous desktop units. The VAXStation 3100, provided users with higher performance and extensive memory. Users could immediately run software written for the VMS operating system. Memory and performance capabilities permitted applications to be shifted without problem.

Follow-on products extended the performance and memory of the workstations as they continued to utilize new advances in technology. Peripherals such as disks have also grown to reflect the growing need for increased capacity on desktop units. There was a continuous drop in the price of storage per character. DEC also expanded the graphics capability of its computers, reflecting the need for professional and business graphics.

In 1985 DEC opened its RISC workstation group in Palo Alto, California. The group was involved with the development of early VAXStations but was also established to test new architectures and establish DEC's presence in the open-systems community. Capitalizing on Silicon Valley experience, the group became a central point for RISC and UNIX offerings. A direct offshoot of this effort was the introduction of DECStations.

DECStations are strategic products that do not use VAX architecture. Entry DECStations are really personal computers based on Intel hardware and Microsoft software. The hardware is manufactured by Tandy Corporation of Austin, Tex. These stations have received additional software to support X Windows and DECNET and participate in networks with other workstations.

DECStations at the high end also differ from the VAX tradition and use RISC technology. Built on MIPS chips,

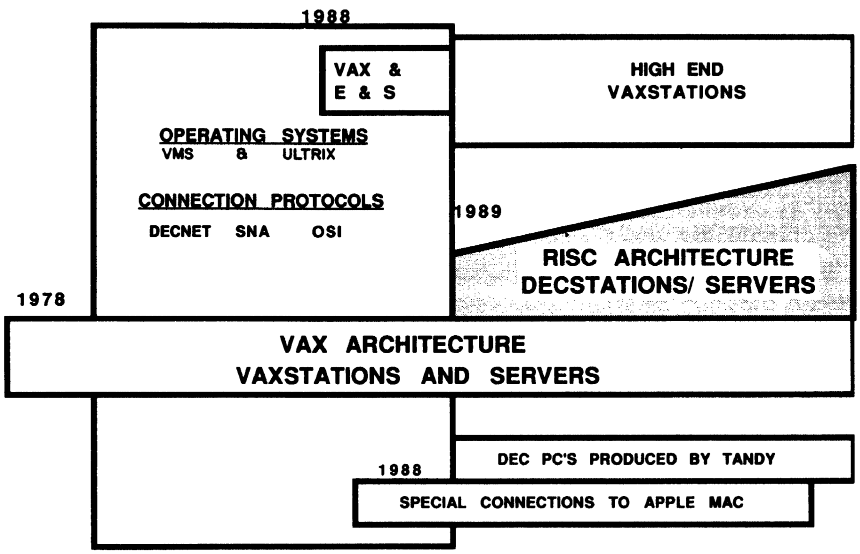


Figure 8.
DEC Workstations

the processors provide different instructions than VAX computers and high-performance levels measured in millions of instructions per second. The selection of MIPS was a departure from DEC's traditional reliance on its own semiconductor plants. Up to the DECStation announcement, the company had been committed to VAX architecture, but rumors of internal alternatives were circulated within the industry. Eventually the decision was made to use MIPS RISC chips; however, the design was modified to support the byte structures found on other DEC architectures.

Using high-performance RISC processors, DEC introduced servers using the MIPS chips. Both memory and performance characteristics were greater than those found on workstations.

With the introduction of DECStation 5000, the company changed from standard bus attachments and introduced Turbochannel architecture. Intended to provide higher performance than existing industry standards, turbochannel architecture is expected to be used by other processors in their line of workstations. DECStations are expected to maintain a competitive pace with other industry alternatives.

DEC has a history of unique contributions to computer technology. Although the company uses existing open standards, it has defined unique industry implementations. As one of the seven founding members of OSF, DEC has provided leadership in defining a non-AT&T UNIX. In another instance, DEC worked jointly with MIT in developing X Windows, which is supported on both VMS and ULTRIX. The capabilities of the window support was demonstrated to the author when VAXStations, DECStations, and personal computers were connected using a LAN. The demonstration showed a ball bouncing on several screens. The synchronous bouncing correlated to the interactions of common user interfaces across the network of heterogeneous workstations.

Another example of DEC's creative approach to workstations is NAS (Figure 9). NAS is a software architecture that expanded DEC's existing protocols to provide information sharing of either data or compound documents across heterogeneous workstations. The architecture has been implemented in routines that reside on DEC's existing system and network designs in a way which enables users to define applications that can communicate between various personal computers, UNIX- or VMS-based processors, Apple Macintosh, and X terminals using the network. Applications using NAS have access to X Windows as well as standard system services such as electronic mail, presentation services, database access, and applications control. The communications protocols adhere to OSI messaging services using standard X.400 or electronic data interchange (EDI) services.

NAS uses the *On Demand* command to provide interactive support in the network, while static support is provided with builder routines. Rated by some analysts as a more open approach than SNA, NAS is an alternative for open-system connectivity.

DEC's proprietary network system DECNET has been popular with users. This system has been used to provide connections for large and small systems alike. Begun in the 1970s, DECNET has passed through a number of transitions that reflect driving forces in the business community as well as different technological advances. DECNET uses personal computer systems architecture (PCSA) to connect personal computers to itself.

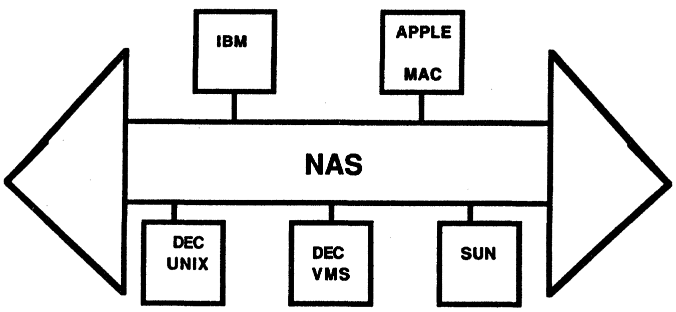
DEC's data interface format (DDIF) defines a database structure capable of handling compound documents. This internal standard is based on the open-document interchange format (ODIF) and open-document architecture (ODA). DDIF supports text, image, graphics, and voice combinations on secondary storage device.

DEC has positioned itself to provide leadership in workstations. Its focus on multiple architectures will help it retain existing users while at the same time bring newer customers to its support levels.

Considerations

The professional workstation market continues to show strong growth as unit prices fall. The three leaders in this market capitalized on this power and have established products for a wide spectrum of applications. Application implementors have benefited from combinations of graphics and processor performance, as shown in Figure 10. Although competition is very strong with the introduction of advanced products from IBM and Data General, the three leading companies are expected to retain their leadership position in workstations in the early 1990s.

Figure 9.
Network Application Services



Open-systems support and, more importantly, the wide adoption of UNIX have helped create an environment conducive to workstation acceptance. Each vendor can enter the market in a relatively short time with a new hardware architecture that uses the standard system.

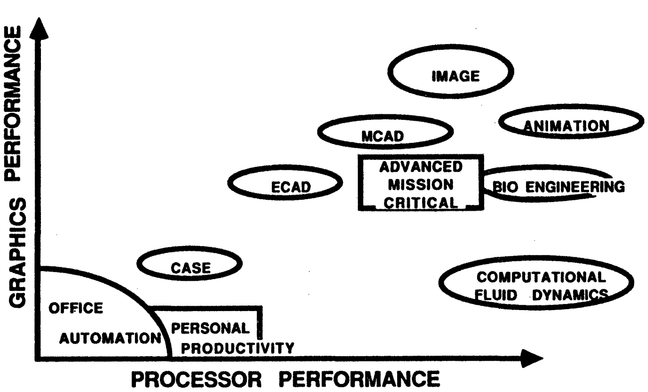
New applications must be designed to capitalize on the ever growing power of desktop computers. The applications will be different than traditional workstation software and will use the full capability of the new technology.

As workstation capability continues to increase, the leading designers will emphasize systems integration as a single focal point for larger users. This will change the characteristics of the competition and place additional pressure on new companies entering the workstation market. Unit sales will be affected by the actions of these leaders and the emerging competition, which is discussed in the next section.

Challenges of the Fourth Wave

One of the biggest challenges of the fourth wave of computing is putting all the pieces together. Workstation products have been successful for individuals and on small networks, but today these products must be connected into a wider network that spans corporations. Workstation products must integrate the company and equipment, as well as the data that drive the company. This section examines several different views of the fourth wave of computing and the effects it has had within different groups of the data-processing community. We examine techniques for applying the technology without upsetting the normal operations of the corporation.

Figure 10.
Applications and Workstation Performance



Five separate views are considered here, as shown in Figure 11. The first view is that of the technologists who consider workstations from the perspective of key advisors or strategists. The second view is that of customer, or user, management, who must justify, install, and integrate the equipment. The third view is that of the company's information systems group, which integrates computer solutions into existing networks. The fourth view is that of the end user who is introduced to the newer computers and must access them. The fifth view is that of vendor management, which must establish the value of the vendor's products in an era of open-systems standards.

The fourth wave with workstations and LANs provides significant opportunities to all users of the technology. Each user can receive direct benefits, but each user faces the challenge of ensuring that the workstation is effectively used.

The Technologist

A few months before completing this report, I reread Tom Forester's 1980 book, *The Microelectronics Revolution.* When wrote his book, microcomputers had just arrived and predictions for their use were bullish. The 1980s became a decade in which their technology was refined and product offerings flooded the market. Today's technology is more powerful than the technology available in 1980, and the potential seems limitless. Increasing microcomputer power is available to all potential users, but this is especially true for workstation implementors. Ideas developed by desktop computer designers will mold successful applications in the 1990s. Mission-critical applications will capture this power and provide a competitive edge to companies aggressive enough to understand and use the capability. New application areas, which can assist companies achieve competitive bottom-line leadership, will use this increased power.

Technologists are faced with challenges to determine appropriate products that they can recommend to management. They search for innovations that can improve a company's efficiency. These decisions occur in an age of rapid change, which challenges both the knowledge and processes used to reach decisions. The technical selection begins with a mapping of requirements against potential solutions, which are derived from an up-to-date knowledge of both the technology and the business. A decision is reached after a systematic consideration of alternatives, using something like the technology-maturity matrix shown in Figure 12. Introductory products are the riskiest, but if they are used wisely, they can provide significant leverage. Stable products are usually the safest, but they often lack the needed edge to move the company in front of its competitors. Emerging products may or may not meet business needs but should be considered as the products mature.

Each technological wave has the four stages of development indicated on the matrix. A new wave usually is in the introductory stage for 3 to 5 years. During this period new technology is introduced and tried using different applications. Workstations have already passed through this stage and have been used in a number of successful application areas.

The emerging stage represents the period when the products are identified with their appropriate applications. This period remains volatile as companies adjust to meet market needs. Initial product offerings occur during this stage and vendors mature. Often specific products or vendors will disappear during this stage.

During the stabilizing period, products are refined with each subsequent release. Growth has slowed, and the number of vendors shrinks as competition increases. Personal computers are in this stage with many companies disappearing.

In the maturity stage the market is determined by the leading vendors. There is no longer any large growth in the size of the market, and introduction of new technology has slowed. New technology does provide additional performance, but the architectures that are used usually remain constant. Users have developed a large repertoire of software that is vital for their corporation's growth. Both minicomputers and mainframes have reached this stage.

A good example of technical maturity are superworkstations. These high-powered workstations contain the leading developments in desktop computers. In their introductory stage, superworkstations appealed to professional engineers for their graphics and computational capabilities. As potential users realized that there was a lack of support packages for engineering departments, superworkstations grew less appealing. Eventually adequate software will make these workstations commonplace to professionals including engineering groups, but until this occurs, superworkstations will have limited acceptance.

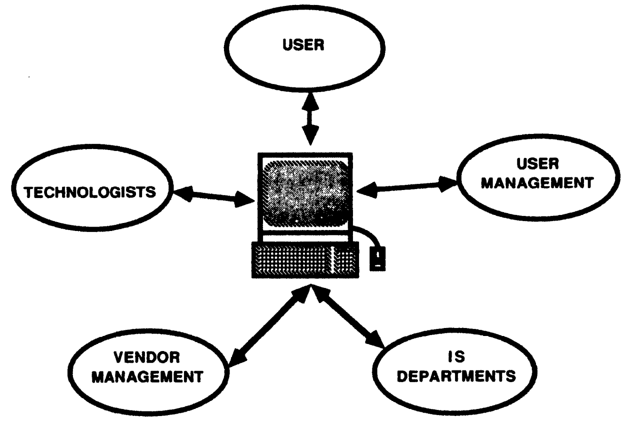
Company advisers often urge the immediate implementation of the latest technology with promises that it will enhance the company's performance or functional competitiveness. Others urge caution to the point that competitors gain business advantages through a timely acquisition of relevant technology. Still others engage in hype to demonstrate their awareness of current developments while lacking a substantive understanding of products' relevance to their business. Management must separate hype from reality and determine when to introduce stable and useful technology.

The effective use of technology can provide business leverage. Yet there is the risk of stopping or slowing operations when the new technology is installed. Business processes are sensitive to change and must be adjusted as modern tools are installed. Managers have lost their jobs by misjudging the maturity or applicability of technology. However, timid managers who have feared to acquire a new technology in time have jeopardized their company's competitive position. Selection is not a black art, since an ordered methodology can be applied, to assess the usefulness of different approaches, matching them to specific business requirements.

The index shown in Figure 12 can be used to assess competitive leverage against maturity. Urgent requirements may need to be met with relatively unproven technology. An evaluation, using the maturity/competitive matrix, assumes that both vendors and their products are assessed. Often a product's success is affected by a vendor's market and support teams. The untimely demise of a product has been known to leave dependent users in a lurch.

A point which should be considered early in the buying process is the way new solutions contribute to a customer's satisfaction Additional satisfaction is equated to retained or expanded business In assessing the advantages of a technology, it is important to understand how the technology improves the company's image and whether this attitude can be translated into additional dollars.

Figure 11.
Multiple Workstation Views



Technologists see the fourth wave of computing as an age that seems to deliver limitless computing power to individual users, yet couples this power in a network across networks. The imagination seems unbounded as opportunities emerge to improve existing business operations and expand the use of computers into new applications.

User Management

User management must determine the correct time to introduce a product or technology. Many companies have appointed a chief information officer (CIO), who is responsible for overseeing shifts in computer technology. The CIO is challenged by new paradigms in the workstation era, including the increased power of desktop computers and the shift from central to distributed processing.

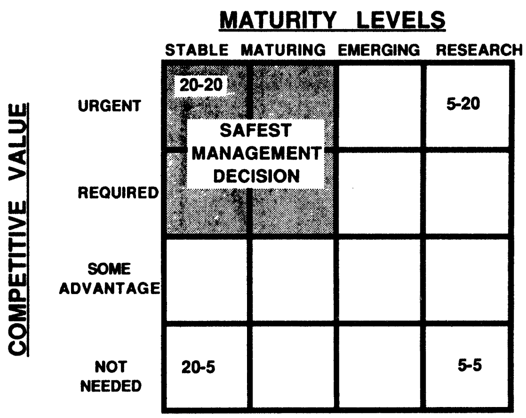
Management's judgment is important in determining the moment to shift and selecting the correct products and vendors. The first challenge in business is justifying new products and possibly modified work flows. It is difficult to acquire newer technology when its relevance is questioned by the business' chief executives. Knowledgeable people question why there is a technological shift and whether it will last. Others question the competitive benefit of the technology or its cost/benefit ratio relative to the existing approaches. Technical questions will arise about the ability to merge the new approach with the proven technical approaches from previous waves.

The company will probably experience a level of insecurity as many experienced professionals are asked to modify their technical approaches. Opposition will come from lack of knowledge that is fueled by the latest trade article attacking the new technology. Increased scrutiny by negative elements within the company will require a much more thorough job by the champion. Also, it is useful to have outsiders brief the company on techniques to ensure success.

Cost/Benefit Analysis

A cost/benefit analysis is the first of a series of required steps to justify the technology within a company. Although technologists become excited about a specific product approach, a new product can often fail to meet the minimum requirements for corporate justification. The manager determines the relative maturity of proposed solutions and articulates reasons for implementing them within the company. An important ingredient in this process is the merit of the approach in achieving competitive leverage.

Figure 12.
Maturity/Competitive Matrix: Maturity Levels



Workstations represent a new technology for most companies and must pass through a rigid justification process. Since workstations and their interconnections exceed the normal purchase limits for department managers, the analysis becomes more thorough and alternative approaches are weighed. The vendor provides the costs, and the internal teams provide the cost/benefit ratios.

Early analysis can expedite approval and ensure the success of installation plans. Change is usually resisted in most organizations. Since workstation proposals involve change, there will often be reactions to and rejection of the solutions. Justification will pass through multiple stages before any solution is adopted.

A source of potential benefit arising from workstation implementation is cost savings. Off-loading of mainframe functions can result in savings, but more often the increased productivity of improved functions will result in savings. Businesses can often demonstrate direct employee displacement, which results in a large savings.

Another justification is enhanced quality that reduces field and internal costs. Since routine tasks can be consistently performed on workstations, they are ideal for producing consistent results. A computer will not often make a mistake, and if the computer does, it is predictable and repairable.

Selecting the right product and vendor is the next challenge for management. In the beginning of a technological wave there are few options, but within a few years, product quantity and quality expand, making vendor selection a complex problem.

Adjustments are necessary in many organizations to handle the increased power of desktop units. Additional work is distributed to the workstation, so different processes are used to handle computing. Throughout the system, benefits can be identified from computing advances. With true scalability, business can expect powerful processing for relatively low costs. Management must adapt techniques to successfully use their computing advances.

Management begins to formulate its plans by reflecting on its short- and long-term requirements. In an era of heterogeneous connections, the requirements can be met by both the network as well as individual desktop processors.

The first test of a workstation's adequacy is its ability to provide functions required by the company. Since many hardware alternatives provide nearly equal capability, it is important to understand the software and network support. Open system approaches have been adopted by companies, but the number of support packages is limited by software vendor commitments. Software vendors develop their programs for systems with the highest projected market share. Management must assess the availability of critical software on the various hardware alternatives, and select workstations with sufficient quality software for the company's needs.

The performance of workstations is changing rapidly. Leading brands may be replaced by the competition in the following month or quarter. Management should select workstations that meet today's needs and have a growth potential. The aggregate performance of connected workstations is expected to exceed that of several supercomputers by the middle of the decade.

Benchmarks have been useful in determining performance characteristics. Popular benchmark series have been used to assess different workstations by focusing on successful instruction mixes. Whetstone benchmarks, which were derived from a large number of engineering and scientific programs, express their results in Whetstone instructions per second. The Dhrystone benchmark is written in C and measures fixed-point performance in Dhrystones per second. Linpack benchmarks are coded in Fortran and reflect intensive floating-point requirements. Results of Linpack are expressed in MFLOPS. Identify the benchmark that closely reflects the required job mix and use that benchmark to assess alternatives. After the benchmark tests are completed, management should have an objective set of numbers to evaluate the alternatives.

Workstations exist in a heterogeneous environment, where installation success is determined by the ability to introduce older applications on newer computers with minimal impact on the existing systems. Incompatible systems can cause major delays in profitable enterprises. There are numerous examples where a lack of compatibility of new technology with existing products became a major deterrent to migration of the existing products. Open architectures provide workstations with significant capabilities, but most older systems have not been reprogrammed to conform to these architectures. Migration involves connecting these older applications and databases.

Another management consideration is expandability. Powerful processing units will increase in numbers in the next decade. To assess the adequacy of the systems, it is necessary to determine their growth characteristics. Will the architecture be scalable only for a short time? Can a particular workstation carry the company through the next generations of processor improvement? Installed workstation systems will grow whether their applications perform well or not. High-speed processing can speed and magnify bad results as well as correct results.

A concept called "softer software" is worth considering. Such software is both easier to use and more powerful than anything available today. Apple Macintosh software created new definitions for ease of use, but even these packages did not go as far as possible within the framework of powerful workstations. Tomorrow's workstations must provide significant new functions over today's products to sufficiently use the additional power. Automatic navigation tools will simplify data access on a network. Additional questions that must be answered include: Does the workstation provide tools for video and audio training packages? Are the application development tools simple enough to develop these processes internally? Is object support inherent in the design of the workstation?

Serviceability is a problem for workstation users. The need to track and correct operational problems in a distributed network has led to the introduction of sophisticated tools. Managers need to consider the tracking mechanisms provided by the vendor and determine whether they connect to existing architectures. The level of the diagnostic program is required by individual users, and how much of it can be automated must be determined. If serviceability is provided locally, determine whether the vendor provides classes for local people.

In network computing, diagnostics must extend beyond the simple determination of hardware and software problems to include network problem determination. These diagnostics should isolate failing components and provide recommended techniques to repair them on the network. Network programs track error activity on the network and can prescribe preventative remedies.

Closely connected to serviceability is reliability. Fortunately integrated circuits have produced highly reliable components, and the breakdown of a single workstation is not as troublesome as the loss of a mainframe. However, the breakdown of several vital network parts, such as servers and the backbone nets, can delay or stop group work. The manager must assess the company's reliability needs and measure the parts of the system against these needs; the reliability record of hardware and software vendors must be determined.

A difficult but necessary assessment is whether the vendor can survive through a strategic cycle. Small companies often offer products that are unavailable from more established companies, but these small firms run the risk of collapsing. Vendors must be selected based on their long-term viability. Since most data-processing products are around for years, management must be confident that the vendor will survive long enough to service the product. Assessment of the viability of the different vendors begins with an understanding of their financial and organizational stability but also includes the satisfaction of their existing customers. In addition, a technological assessment determines whether the company has management depth to ensure their continued performance.

Management is faced with new challenges in selecting workstation and LAN vendors. Management must apply simple ground rules for the selection process and establish a checklist that includes the items listed above to ensure that a vendor meets the company's specific needs.

Information Systems

During the personal computer era, management information systems increasingly supported departments that used the available computers. *MIS Week* estimated that over one-third of all MIS personnel were reporting to these departments by 1990.[3] Increasing autonomy resulting from desktop computers has enabled departments to bypass MIS altogether. With the introduction of workstations connected through LANs, there is a renewed need for the professional services of information systems groups. The logical center of processing has shifted from the glass house to departments that use these workstations.

The pragmatics of workstation technology raises the question of how to integrate workstations into the existing data processing. New information is created on workstations using data that must be incorporated into the corporation's databases. Workstations and traditional terminals create different work flow patterns. A step process ensures an easy and logical transition from the older processing technique to the new one.

Integrating the workstation into active computer operations is a challenge. While many existing systems are preconfigured to handle terminals, they must be reconfigured for workstations. Mainframes and minicomputers have used static system table generation techniques for adding computer capacity. Workstations require dynamic connections and shifting locations so they have adopted more dynamic generation processes. Existing database and communication protocols used to connect mainframe computers will have to be extended to handle client/server processing.

Another challenge is managing complex workstation networks. Synchronization of resources is needed in a groupware environment. Connectability is a major problem that needs to be addressed. Companies establish their own way of connecting networks that is known to the IS group but not to external integrators. Support for different attachments is widely claimed, but often establishing that support requires work from the departments that use the network.

Many vendors have advertised that their products are integrated systems, but again the reality is distant from the claims. Some specialized areas have been able to have large portions connected together into a single system. An example in electronic design is the integrated approach of Mentor Graphics to electronic design in which mission-critical applications are integrated into comprehensive packages.

Integration costs, measured in time and equipment, may make the acceptance of new products prohibitive. Integration can be considered the third dimension of the maturity/competitive matrix, as shown in Figure 13. This integration represents the combination of the technical solution and the organizational solution. Processes currently in use must be shifted to include workstations and LANs.

The IS group has emerged as a pivotal resource in installing the complex workstation/LAN combination. The group's technical skills are important in identifying and solving problems. The group must form part of the team that makes the workstation a competitive part of the business.

Workstation Users

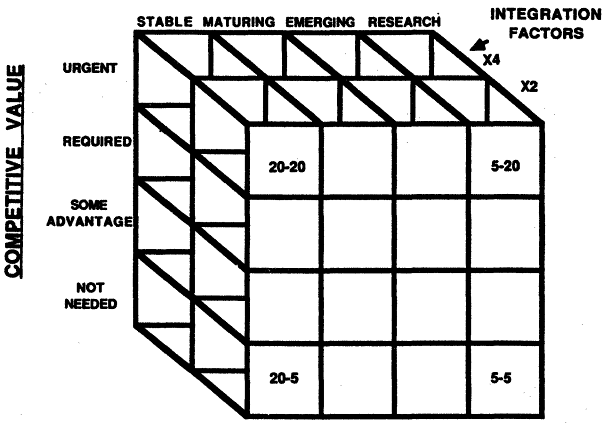
The workstation user is the first to recognize the value and potential problems associated with workstations. Often users come from existing mainframe environments and are now exposed to complexities that were once handled by a room full of operators.

The first opportunity for reaching the user is the interfaces. There has been a continual shift from the complex screens created on mainframes and downloaded to terminals. Today's screens have menus that are simple to use. Keyboards have been replaced by mice, which simplify the entry process.

Simplicity has been extended by the implementation of software based on the messy desk concept. Through the use of windows, the desktop computer user has an environment similar to a desk. Each window contains an active task that completes a different job. Thus, one window can contain the word-processing program, another a spreadsheet, and a third a mission-critical application.

New dimensions in power provide opportunities for even further breakthroughs in user interfaces. Bit map screens on inexpensive workstations provide advanced graphics capabilities. An emerging application area is visualization, which is the presentation of nongraphic data in a

Figure 13.
Maturity/Competitive Matrix: Integration Complexity



graphic format. It is possible to extend this capability beyond scientific tasks to commercial applications. Stellar Computers introduced scientific visualization on very high speed workstations to produce improved presentation systems for users.

Vendor Management

During the past decade workstation vendors have witnessed dramatic growths in their markets. In the next decade vendors will define unique qualities that make their products different. This uniqueness challenge is partially driven by the wide acceptance of standards and the rapid change in markets. Uniqueness has become more difficult to define because many workstations have identical functions and perform equally well. Performance characteristics change with each new technological advance, which makes some companies leaders for only a brief period. UNIX has been adopted as a standard operating system by most vendors, so even this level of support is common.

As prices have tumbled, traditional sales channels are being replaced by commodity distribution techniques adopted for personal computers. Until the personal computer era, most computers were sold through direct sales contact with the appropriate information managers. Low-cost hardware and software made such approaches unrealistic, since the cost of sales had to be contained within the product budget. To reduce overhead, personal computer companies adopted direct retail or even mail-order approaches to sales. Initially the computers were sold over the counter at computer shows or fairs, but computer stores were opened and offered additional services such as education.

Pragmatics drove larger companies to use comparable sales outlets. The majority of IBM personal computers were sold through stores. Chains such as Sears, Computerland, and Businessland developed their computer skills and provided additional services. Education and service became prime ingredients in successful sales strategies. Service was performed in local stores to reduce costs, and hot lines were installed for rapid customer response.

Professional workstations are traditionally sold by dedicated sales people who contacted scientific and engineering managers. As the market broadened to commercial applications, a similar approach was used, but this required additional skills from the sales force.

Workstation companies have turned to resellers as the price per unit tumbled. The first group to be approached was the value-added resellers (VARs), who provided additional software or hardware to reach niche customers. As prices of workstations fell below those of personal computers, vendors adopted new approaches to retain their profit margins.

The large company is an important customer, and unique approaches have to be adopted by any vendor seeking corporate business. This begins with active support for system design and continues to full integration of a system solution. New expertise will be required to compete in this business. The vendor must acquire the system skills of those familiar with mainframe connections and the necessary communications protocols that make these solutions successful.

The large company will also expect discounts and service agreements that are related to the purchase volume. Many mission-critical applications must not fail, so products must have sufficient redundancy to ensure their successful operation. Spare workstations will be required by contracts since the company cannot afford to have any of its staff not working for a long period of time.

Software vendors are either porting their personal computer products to workstations or creating new applications using the power of the workstations. Vendors exploring groupware are forming alliances or mergers with companies that can provide network software.

Competition will continue between high-end personal computers and workstations. The personal computers bring a richness of individual applications, while workstations provide platforms that connect to larger networks. Workstation vendors will play an ever-increasing role in complete computer solutions. Traditional science and engineering customers will continue to expand their usage of workstations. Even nontraditional markets such as financial institutions are finding the aggregate power of the workstation attractive for new applications. Competitive workstations will emphasize the use of more powerful functions such as high-resolution color graphics, macros, and support for other databases. Vendor success will depend on the resolution of many difficult problems as workstations continue to compete directly with medium- and high-end personal computers. One of these problems is defining the right applications to use the power of the desktop unit.

Harnessing Workstation Power

The potential power of a workstation is projected to be greater than the power of today's mainframes. Each new generation of chips increases its processing capability beyond those of previous models. With projections of 100 million instructions per second for less than $5,000 by 1995, many business lenders fear that this great power will lie idle on desktops. Certainly many personal computers with much less power are sitting idle on desks, while their support bills mount. A brief survey of many corporations will probably find that only a fraction of the large amount of desktop computing power is being used.

The answer to this problem is not simple, since personal computers contain sufficient computational capability to run most of today's applications. Spreadsheets and word processors only consume a limited fraction of the computing engine. Yet business-critical applications could benefit from the increased computer power found on workstations. Many functions still exist that cannot be run on the computers available to management today. Apart from new applications, the first effective use of this computer power is robust user interfaces. These interfaces are beyond the Macintosh's capability, even taking into consideration Macintosh's advanced graphics and imaging.

In today's computing environment, cognitive functions are usually handled in larger computers. As workstation processing power increases, cognitive primitives will be built directly on workstations. These primitives will be very simple at first and will use expert system paradigms. Provision can be made for work-assistant programs which can help professionals such as engineers and architects make correct decisions. An engineering assistant can provide the practicing professional with automated checking of design assumptions. Comparable assistants can be produced for all disciplines, including management in the form of an executive information assistant. Modern executive systems will benefit from the use of comparable cognitive interfaces to navigate through operations reports in executive information systems.

Users of powerful workstations will benefit from advanced data handling. This capability will use object-oriented technology to provide additional attribute information about the system's information. Most applications in this era will either directly use object-oriented databases or will eventually scaffold objects on top of relational database structures. Effective object handling will require the functions and power found in today's processors. *Data navigation* will provide the ability to search and find information in the company data structures. Necessary bridges and conversions will be built between different data types to provide the correct protocol for timely, interacts work.

Connected to data navigation are *data mining* routines, which determine important data in very large databases. Using preestablished criteria, data mining routines search combined databases for critical information that can shortcut extensive searches for processing elements. Data mining will provide input to advanced graphics routines in the form of *visualization.* Flat-file data will be presented in graphic images to help workers make critical decisions rapidly.

Advanced cognitive support will provide local capability for simple rules on a workstation. An easy-to-use language, perhaps a natural language, should be used to make these cognitive tools effective for the unskilled. Many developers have suggested that a formal programming technique should be used for this process.

Transaction handling facilities should be transparent to end users. Such support already exists in mainframe products but has also been implemented in computers that use the OS/2 Extended Edition communications manager. In addition, the ability to switch between voice and data transactions will become important for many workstations. American Express agents, for example, handle both voice and data transactions at their workstations. When transactions are shifted to other agents or supervisors, the voice connection is also shifted.

Increasingly, multiple functions are performed through a single workstation and require parallel management routines. These and many other hardware capabilities offer challenges to software manufacturers. Hardware has again provided an opportunity for software developers to develop powerful applications. Untapped hardware capabilities exist in interactive video and FAX support. It is clear that hardware power will continue to increase, and it is expected that software developers will take advantage of this new power.

Enablers and Inhibitors

Managers will encounter enablers and inhibitors to the introduction of workstations. It is important to recognize and understand some enablers and inhibitors that can be encountered in the company.

One enabler is the price of computer power. Personal computers have established a tradition of inexpensive power, and most managers have come to expect it. The broad accessibility of personal computers is familiar to many people. Executives have met peers who have shared stories of their success with workstations. It is clear to many managers that computer power can be used to move higher quality products to market more rapidly.

Another enabler is the reduced price of desktop units. Workstation prices have fallen to within a few thousand dollars of those of personal computers, and this has, therefore, become an enabler. The trade-off between a few thousand dollars and improved productivity is easy to predict. Similarly, the low interconnection costs are obvious to companies looking for system alternatives.

There are also inhibitors that slow workstation acceptance. The first inhibitor is the organization and individuals who resist change. Many changes using high technology have been rejected by a group of people who resist the change to their traditional approaches. These individuals question whether a technology is sufficiently mature for implementation or whether the vendors will survive. A second inhibitor is the workstation's additional hardware and software complexity. Support for workstations is expected to be more difficult than for personal computers.

A third inhibitor is the higher prices of workstations and LANs. Although these prices are tumbling, they continue to be higher than those of personal computers or of alternative terminals. Increased function must be realized before there is a general thrust toward workstations. UNIX is another inhibitor since it is difficult to use and program. Although much software is available to its users, many installations are reluctant to enter this complex world. UNIX has also been noted for its security penetration and the shortage of trained programmers and trouble-shooters.

Considerations

The fourth wave of computing promises an exciting end to the century of technology. Today's workstations already contain more power than most mid-1980s mainframes. Standards are pivotal to success in this wave, yet implementations have trailed the standards committees. Often the standards are the result of compromise, comparable in some cases to defining an elephant by a committee. A challenge for the coming decade is to obtain conformity to the agreed standards and identify areas that will further simplify the integration process.

Challenges exist for managers and professionals as they implement fourth-wave solutions. Increased power on workstations is a positive factor, but this same increase in capacity raises questions of its effective usage. Many individuals in a corporation will challenge the need for such power within the framework of the company's needs. Many managers will be comfortable with the status quo and fail to understand the competitive advantage in introducing the new technology.

Users and managers require a higher degree of computer literary to understand the complexities of the new wave of computing. Understanding the various elements of open systems will require knowledge not usually available in many companies. The demand for individuals trained to use UNIX will increase during this period as UNIX becomes essential for all companies shifting to fourth-wave computers.

Central processing avoided the distribution problems of the new operating systems and their supporting applications. As these applications and data are distributed, new pressures to manage the network will emerge. Mission-critical applications are even more pivotal to the completion of successful installations, while shrink-wrapped software is necessary for general user support of generic applications.

Integration of the computer facilities will become necessary for companies that successfully enter the fourth wave of computing. Systems integration skills will be important for the companies and individuals who seek to make an impact on this wave. Alliances of companies that have perfected these skills will be very important.

We expect to see increasing use of strategic alliances between critical vendors to support expanded capabilities. Companies that were once competitors will find themselves working jointly to produce products in this wave of computing. There is still plenty of room for new vendors to emerge in both workstation hardware and supporting software, but they will have to find a role in the heterogeneous networks. Client/server architecture will become better defined as new products are announced.

Applications increasingly require access to shared data from different computer sources. With the power threshold rising on workstations, additional data can be distributed across multiple desktops. Applications are becoming available on hardware from several vendors providing lower cost access to distributed data. Since the users have many low-cost alternatives, vendors have been forced into providing real value on a timely schedule. They must also provide applications which provide added value in distributed environments. Differentiation between workstations is based on the quality of the products, the vendor's presentation, service and support, business relationships, and access to required applications.

RISC processors will show continual growth and eventually will be the prevalent hardware architecture for all workstations. It is expected that by the end of the century these processors will handle billions of instructions per second. Disk storage in the processors will be placed at billions of bytes instead of the millions found today. It is becoming clear that processors using RISC architecture will dominate the 1990s, but we can also expect to find the previous generation of personal computers participating in this power wave. Specifically, high-end personal computers and Macintoshes will be key participants in this developing market.

The third wave has created many of the good ideas that are being absorbed by fourth-wave vendors. We have many years before this wave will reach its apex, but during that time we can expect to see the first appearances of the next wave. Hopefully all the readers will be there to welcome it.

Bibliography

[1]Bach, M.J., *The Design of the UNIX Operating System,* Prentice-Hall, Englewood Cliffs, N.J., 1986.
[2]Kernighan, B.W., and Ritchie, D.M., *The C Programming Language,* Prentice-Hall Inc., Englewood Cliffs, N.J., 1978.
[3]Goff, L., "MIS Experiences Darwinian Evolution," *MIS Week,* May 21, 1990, pp. 22-24. ■

Executive Support Systems

In this report:

Datapro Summary

An Executive Support System (ESS) is a computerized system geared for corporate- and division-level managers. This report provides an overview of ESS: it offers an explanation of such systems, describes the basic classes, and focuses on the key benefits of an ESS. The report also identifies eight critical factors in ESS implementation.

Definition of an Executive Support System

An Executive Support System is a system which:

- Centers on the use of computer resources.
- Is used by the top two levels of management, either at corporate or the division level. These strata of managers think differently. They operate in a different, more abstract world. Middle management has different, more predictable information needs.
- Is applied to any business function. This may be monitoring, analysis, or communications, such as electronic mail, which has changed the way many executives look at the computer.
- Is used hands-on by the top-level managers or by their staffs. When executive questions come up, analyses should now be made which were previously too difficult.
- May be designed for either individual or organizational systems. They may start with individual use then expand to organizational use when many questions arise, such as ownership and political issues.

There are choices to be made in ESS (Executive Support Systems) design and application. Ultimately, ESS involves choices about what is important to the executives and to you as the developers. What kind of message are they trying to send to the organization? How do we use information? What information is important? These are often implicit choices that are not verbalized, but the choices must be given thought. By

This Datapro report is a reprint of "Executive Support Systems: The Myth and Reality of Top Management Computer Use," by David W. De Long, pp. 142-149, from *Journal of Critical Issues in Information Processing Management and Technology*, Volume 7, Number 4, 1990.

choosing to focus on this set of data in which these are the key factors or to design the ESS as an elite system with access only to certain people, we are sending out messages about control over the use of information.

ESS can be considered as an umbrella term covering all executive computer use. Executives fundamentally are doing two things with computers:

- Information retrieval, data manipulation, and analysis
- Communications

Most of the vendors in this area are essentially selling information retrieval applications. Some of the major vendors, such as Pilot and Comshare, are now starting to point out that electronic mail is an important part of their application. A brief comparison is:

- Executive Information Systems (EIS) are principally information retrieval.
- Executive Support Systems include information retrieval and communications. These systems are never just electronic mail.
- Executive Decision Support Systems (Executive DSS) cover the modeling of semi-structured decisions that tend to be repetitive.
- Senior management makes very few of those types of decisions. Merger and acquisition analysis may be one area.

The design of ESS should reflect four important dimensions. They should never be "individualized" to the point where these dimensions are not considered. They are:

1. The personality and cognitive style of the executives.
2. The executives' philosophies about the use of information. Are they really interested in emphasizing control? Are they interested in getting a better sense of what is happening? Do they want surveillance? Are they sharing information and broadening the exposure to information?
3. The job level and function may require very different information needs.
4. The business situation is key to the design of an ESS.

There are a number of myths associated with Executive Support Systems. These are in no particular order, but they should be watched as they may be built into our assumptions when we are planning an ESS. These myths are:

- **All executives want the same kind of system.** All executives do not want the same type of system. They have very different needs. Do not start with a technical solution, such as a package program, and assume that it will fit all needs.
- **We just need to solve this technical problem.** Forget this myth! ESSs do not live or die based on the technology that is chosen. You must focus on the business issues.
- **We have to cost justify an ESS.** If the executive wants the system, you will get the resources and the data. Cost justifications may be required, but in most cases they are not.
- **All executives should be using computers.** This is nonsense. Executives vary widely, and it all depends on whom you are dealing with. Executives should have computers who really feel that there is benefit in them.
- **There is only one way to implement an ESS.** This is also nonsense. There are many ways, many programs, and many approaches; the best is the one that works.
- **The system should always be pre-sold.** There are no fast and firm rules. Some are strategic and some are not. They are developed in many ways. It all depends upon the political environment in the company.

Classes of Use of an ESS

There are several classes of use of an ESS. A class of use is how the executives are really using these systems. It is a framework for thinking about the types of applications. A categorization of the classes of use is: communications, monitoring (either internal data or external data), analysis, and office support.

Communications means considering a system that allows the sharing of data between different individuals, offices, or sites. In a manufacturing business, for example, senior management may want to look at inventories, plant loading, quality, and cost issues. It may want to share this information with international sites. If an executive starts to use an ESS for communications, the organization gets the message quickly that this is the way that the executive wants to communicate. Electronic mail, for example, is a cultural thing. It works well in some companies but not in others.

Monitoring is by far the biggest application area. Frequently, management wants to look at predetermined, pre-formatted data in many systems, particularly external data. It may want to look at internal data, such as plant operations. By putting in such a system, management essentially shrinks the "laundering" of information by at least one time period. For many executives, external data is an important area of interest, particularly as the available databases get better. Usually that information needs some synthesis before it is really usable.

Analysis varies considerably in executive use. Only about 20% of senior management systems have any analytical capability. Many do not even like spreadsheets, while others are expert at handling "what ifs." Some like to analyze capital spending patterns. Others like to analyze potential candidates for acquisition. There are many variations, but those using analysis are the exception.

Office support includes many applications, such as word processing, automated rolodex, calendaring, and so on. Again, these are unique with the individual executives. These may appear to be minor applications, but they should not be overlooked. Do not focus your attention on them, however, unless executive interest is expressed.

Key Benefits of an ESS

The benefits of an ESS evolve as the impacts of these systems evolve. Unfortunately, many studies have agreed that at least 50% of the ESS systems developed either fail or are very mediocre. Another large number only provide simple benefits, such as increasing efficiency, or allowing us to use electronic mail to get more timely information. They may provide a better sense of control over the organization, but they have no big impact. About 20% of ESSs provide real, major advantages. These are really important and include those systems that:

- **Force executives to re-think their information needs.** Executives are not used to looking at information from a top-down perspective. Historically, they have received information as a by-product of the existing reporting systems. Now they can consider what really matters to them.
- **Enhances executive understanding.** If the executives consider what information really matters to them and how the delivery system of the information should be designed, there is a tremendous benefit from the mental exercise involved. It enhances the executive's understanding of the business. This is not really tangible, but a number of executives have said that the computer has given them a better grasp of the external forces and of the specifics of the internally-generated financial data. If you try to build a system that enhances executive understanding of the business, the system will fail. Many executives who have used an ESS, however, have said that they have received such a benefit.
- **Improves management processes.** Some executives are pleased that they can cut the data 17 ways and explore a great variety of options. An effective system is difficult to implement, but once implemented, it can really make a difference in the way the organization is managed. Such executives feel that they have a big advantage in dealing internally with their business units, and it changes the way that they make strategic plans.
- **Communicates philosophy: learning vs. control.** Possibly the key benefit of an ESS is the philosophy that you communicate to the rest of your organization. Is the system a control that tries to keep everyone in line? Is the system a tool to teach people what is important, and what should be managed? If it is the latter, there is a good chance that it will be successful and will trickle down through the organization. These are actually power issues. In the long term, increasing control is not going to provide a competitive advantage that is sustainable. It is better to use information to create change to stay ahead of the competition. Control has little tactical advantage.

Ask yourself and the executive you are supporting what you are trying to change. Determine what

Figure 1.
Critical Factors in an ESS Implementation

- Committed and Informed Executive Sponsor
- Operating Sponsor
- Clear Link to Business Objectives
- Appropriate Information Services Resources
- Appropriate Technology
- Management of Data Problems
- Management of Organizational Resistance
- Management of Spread and System Evolution

value you are adding to both the individual executive's effectiveness and the organization's effectiveness.

There are thus a number of questions to consider in deciding on an ESS. These include:

- Is it the most useful methodology?
- Is there sufficient structure and planning?
- Is there sufficient time and access to executives?
- What are the politics of changing information flows?
- What are the technology constraints?

ESS Implementation

Harvard Business School has identified eight critical factors in ESS implementation. These are listed in Figure 1.

The need for an executive sponsor is one that distinguishes ESS from operational information systems. It should be the most senior user, preferably the person who makes the initial request for a system. The executive sponsor should initiate the development program, oversee the development, and communicate continuing interest as the work progresses. This person should have realistic expectations of the available technology and data that can be accessed, and should set realistic deadlines for the development of the ESS.

The operating sponsor should be a trusted subordinate of the executive sponsor, who knows both the business and the technology. The operating sponsor must be completely involved in the project, even when the executive sponsor is involved down to the detail level. The job of the operating sponsor will depend upon the complexity of the ESS application and on the skills and experience of the I/S team which is working on the project. The operating sponsor will be involved to a great extent with problems of data ownership, data access, and data content, as well as with the administration of the project. Frequently, the operating sponsor is the executive assistant to the executive sponsor. Other common candidates are the Controller or the CFO.

There is always a problem of linking the technology to business objectives. In the first place, few in top management can say in advance and in any detail what they want in an ESS. There are several reasons for this:

- They may not be particularly interested in the specific problems that can be solved most readily by the technology.
- They may want to look at an ESS, but will not have clear objectives as to where they want to go with it.
- They seldom think about the business as it relates to what information is available and what reports can be produced.
- They do not really understand what the computer can do for them and how far they can go with computer solutions.

There are more fundamental problems that senior executives have with ESS computer solutions. They are asked for a set of requirements that will be relatively fixed, yet their needs and objectives are constantly changing. Their thought process is seldom the rational, step-by-step computer process but is more linked to feelings and ideas. Even the term "business objectives" has a variety of different, changing meanings to them. In their decision process, there are other forces that dominate, such as the culture of the organization and the available, proven technology.

The problem of appropriate I/S resources is usually solved by the level of the executive sponsor. Normally, resources are made available for high-level sponsors. The difficulty comes when trained staff are not around that can handle such a project. An ESS is frequently too sensitive for using an outside consultant. Often, the accounting firms sponsor such efforts, however, and have the trust of management.

The problem of appropriate technology is seldom a limiting factor today. There is a wide variety of easily-used hardware and well-tested software for ESSs available on the market. It requires constant study to stay on top of the possibilities.

The management of data problems is usually the most serious obstacle that is faced. Generally, the data that is required must all be available on a relational database that is well-integrated into the routine company operations. There are numerous problems of data ownership, data control, regular data updating, data verification, and data security that must be solved. One such problem is that of data validity. How can controls be established so that analysts who are working for the executive sponsor cannot simply "adjust" the data, for a variety of reasons, until the data in the executive's output does not match the controlled data in the operational files?

The **management of organizational resistance** is a complex problem that is unique to every organization. The development of an ESS for a senior manager is a political problem and the potential resistance is great. It may be:

- Staff groups who are performing such analyses
- Line managers and their departments
- Other executive users who have somewhat different data files
- The CIO and the I/S staff who see their control of the information getting away

The resistance of the line managers and staff groups is directly proportional to the amount of change that will take place in the handling of the information. They will naturally feel uncertain about their new, changed roles and not know what the real objectives of the ESS are. They will feel vulnerable to second guessing by the executive who previously accepted their analyses. There is also the question of data ownership. They see a small I/S group taking over effective use and presentation of their data as they lose control of what is given to upper management. In their minds, the executives will be accessing data out of context. This means that the system will cause them more work and trouble and more detailed attention by the executive. They have no incentive for such change.

Organizational resistance to ESS development is usually passive but profound. There will be ways of slowing the development at every turn. The possibilities of getting value-added will be markedly reduced. Negative feelings will be built up about the ESS work. This means that organizational resistance must be recognized, considered, and met. Ignoring it or forcing the change is seldom productive. It is generally most useful to manage the change:

- Take a political approach from the beginning.
- Rely heavily on the sponsor to make the change palatable and the project visible.
- Build benefits into the system for the suppliers of the data.
- Advertise the participation of the data suppliers on the screens.
- Point out the benefits to other individual users.
- Build relationships and credibility steadily.
- Have discussion meetings with both the sponsors and the resisters.
- Provide ideas and techniques, and let the executive sponsor take actions.

If all else fails and the senior executive sponsor feels that the system is absolutely necessary for strategic and survivability reasons, then let that executive force the change and move on. If this happens, it is important that all the above ways of managing the change be seriously considered and used. Otherwise, there will be no long-term future for the system.

The management of the spread and evolution of the ESS are related to the political atmosphere but are fundamentally problems of good management. Plan. Work the plan. Report accomplishments. Select hardware and software that can be generally used. Keep reporting on and discussing the project with other groups. Work systematically, and do not overload the available staff. Start training programs so that many others in other groups can become equally adept at developing similar ESS applications.

The keys to success for an ESS are the effective links of the ESS to the executive's needs. Follow the thought process of the executive. Some executives like the Critical Success Factor (CSF) approach, with progress reported against the CSF. Others prefer a series of interviews. The most successful technical approach to determining an executive's needs is the use of prototyping. If a series of prototypes are built, shown to the executive, and adapted as understanding is gained, the result will be very close to the executive's expectations. Some important points here are:

- Focus on adding business value with the information availability.
- Manage the expectations of the executive. Do not promise too much, too soon.
- Keep the pilot small, and take time to do it right.
- Build a data infrastructure and a data supplier infrastructure. ■

EIS in the Public Sector

In this report:

Datapro Summary

Progress in the area of executive information systems (EIS) has primarily occurred in the private (corporate) sector, with far less movement in the public (government) sector. Although there are some inherent problems with implementing EIS in the public sector, successful systems can be built. This report examines the differences between public and private information systems, and provides a case study of an EIS developed for a large state government agency.

Private vs. Public Systems

We begin with a question: why has most of the progress with executive information systems been made in the private sector (or in corporations) rather than in the public sector (or in government)? We outline only a few reasons here.

Systems Cost and Risk

It is clear that executive information systems are expensive and not easily justified. The reported cost of typical private sector system varies between $1 million and $2 million for the hardware, software tools, and development effort. Many private sector (corporate) users do not consider this an unreasonable price for something that helps top management make better decisions. According to Richard Scurry of Chemical Bank. "These guys make a lot of money. Something that gives them better access to critical information is worth a lot".[1]

In most public sector organizations the situation is quite different. The cost of the system cannot be ignored. Projects with seemingly vague objectives like the typical EIS are much more difficult to justify than projects such as transaction processing systems, which usually have very well-defined objectives.

Public sector organizations operate with fixed budgets and have little leeway to shift dollars from one category to another. In these circumstances, a risky project like an EIS, often intimately connected with one champion or a small group of administrators, is not likely to show up high on the MIS priority list.

These observations, and others that follow, may apply to segments of the private sector as well. Some firms may be quite cost sensitive and risk

Reprinted by special permission from *MIS Quarterly*, Volume 14, Number 4, December 1990. Copyright 1990 by the Society for Information Management and the Management Information System Research Center at the University of Minnesota.

conscious, perhaps even more so than some government organizations.

Measurement and Organization Culture

In a general way, and with obvious exceptions, corporations can be described as active, while public sector or government organizations can be described as passive. In most corporations there is a clear reward for activity. In the public sector there is frequently no reward for activity. Even if there is, there is often no reliable way to measure output and productivity. In conversation, a senior manager of a large New York state agency made the point: "We don't handle outcomes well. We count anything that moves but do not measure how well it moves. Instead of just asking, 'What do we have to do?' we need to ask, 'How well did we do it?' "

Accountability and performance measurement differences lead to quite significant differences in managerial behavior. For example, if a consumer goods company manager overspends the advertising and promotion budget in a given quarter but produces twice the anticipated profit, the action would likely be rewarded. In contrast, a social services officer cannot exceed the budget even if there is a real need to provide benefits to more families than anticipated.

Data

In both the private and public sectors, data drawn from internal transaction processing and MIS systems may not fully serve EIS needs. For example, any evaluation of the impact of promotions on sales requires data on the timing of promotional events. The accounting system will contain information on when the bills for the promotion were paid, not when the promotion ran.

In addition, public sector data frequently originates from a regulatory requirement that is unrelated to the programmatic activities of the organization. An example from our own university (State University of New York at Albany) demonstrates this point. We know how much we spend for travel to the nearest dollar, for in-state travel, for out-of-state travel, for international travel, etc. But we have no idea of how many of those travel dollars were spent for faculty research, for faculty recruitment, for student recruitment etc. In other words, our system is bound by its categories. Creating new categories is possible only at the manual level; this is therefore not a frequently used option.

Decision Making

Measurement problems in the public sector contribute to differences in decision-making behavior between the public and private sectors. The lack of a clearly defined *bottom line* in the public sector leads to a focus on inputs and budgets, not outputs and productivity measures. Thus, meeting the budget becomes an important output measure, not just an input tool as in the private sector. Further, several surrogates for output are tracked independently in the absence of a mechanism to synthesize a single measure of output. This makes it difficult to allocate scarce resources among conflicting needs.

With more constraints and unknowns in the public sector process, it is not surprising that more parties are involved in decision making. More decision makers, more constituents, and more criteria and values, combined with chronic budget constraints and a lack of enthusiasm for expensive, uncertain projects, make public executive information systems difficult to develop and justify. Is it any wonder that the private sector is way ahead with this new technology?

EIS—State of the Art

Executive information systems have come a long way since 1976 when Ben Heineman, chief executive at Northwest Industries, began to use a terminal and database to monitor and plan the growth of his nine operating units.[2] Yet the progress has not been uniform or steady. A 1983 article in *Fortune* commented:

> The most important factor keeping the computer out of many executive offices is the realization, sometimes barely conscious, on the part of managers that this technological wonder has, as yet, little to offer them. The nature of their work—in a word, unstructured—is such that it's not particularly susceptible to computerization.[3]

Our observation of executive information systems in many organizations can be characterized by some common features that were identified in an early study by Rockart and Treacy[4] and are still true today. These include the preparation and delivery of periodic paper reports that contain heavily aggregated data. Usually the reports are received by the executive well after the data is "fresh" because of delays in gathering the raw data from several sources and processing it in the desired form. The resulting reports are presented in a fixed, unvarying format, usually quite voluminous.

Given the nature of the support provided by the typical executive information system, the executives who use them must still rely on frequent oral reports, anecdotal data, and analysis, if any, carried out by staff personnel. The formal EIS thus becomes only a part, and frequently a rather small, unimportant part, of an executive's total managerial activity.

But this situation is changing. A quote from a 1989 article, again from *Fortune*, highlights the change:

> Duracell CEO C. Robert Kidder manipulated a mouse attached to his workstation to search for data comparing the performance of the Duracell hourly and salaried work forces in the U.S. and overseas. Within seconds the computer produced a crisp, clear table in colors showing that the U.S. salaried staff produced more sales per employee. He asked the computer to "drill down" for more data, looking for reasons for the difference. By the time he finished browsing, he had determined that Duracell had too many salespeople in Germany wasting time calling on small stores.[5]

The Duracell example demonstrates what is needed in contemporary EIS systems: access to a *broader* sweep of data, often at an *increased* level of detail, with the ability to access and analyze this data on an *as-needed* basis. We return to Ben Heineman of Northwest Industries for a succinct definition of what a good EIS should do: The aim of the system should be to provide information to the executive such that he or she can "ask the right questions and know the wrong answers".[2]

Since the development of the Northwest Industries EIS, several other excellent systems have been developed and reported in the literature. The vast majority of these applications have been in the private sector. A good example is the Lockheed-Georgia MIDS System.[6]

This report presents the results of an ongoing project to develop an EIS for a large agency in the New York state government. The system is being used in different and creative ways, leading to a change in the organization's culture, with implicit and explicit impact on the focus of the organization and its measurement systems.

EIS Issues

Level of Detail

Conventional wisdom suggests that senior executives need access only to highly aggregated, summarized data, but our experience and current developments in EIS suggest otherwise. Giving an executive the ability to access detailed data, as in the Duracell example cited above, can give him or her the opportunity to signal to the organization that business is not "as usual." With access to detail, and a good understanding of it, an executive can change the behavior of people at lower levels of the organization.

As an example, the president of Banco Internacional, a Colombian subsidiary of Citibank, forced his entire organization to become more marketing oriented by constantly monitoring customer information and holding his employees accountable for being informed about customer activity.[7]

Access to detailed data at the top of the organization can actually facilitate the pushing down of decision making to operating levels. Phillips Petroleum, for example, gives senior executives access to price information from 240 motor fuel terminals. Such information was formerly available only on various bits of paper. Now that top managers can monitor all price information, individual pricing decisions have been delegated to local terminal managers.[5]

Another important reason for providing executives with detailed data is that summarized information about the "bottom line" is not enough. The ability to "drill down" to the real problem, as cited in the Duracell example, is possible only with specific, detailed data. Former New Hampshire Governor John Sununu, now the White House Chief of Staff, put it well: "It is my conviction that one needs to go down to the lowest source to get intimate, unbiased data".[8] The CEO of Frito-Lay echoes this sentiment: "DSS gives us the information we need—not what someone wants to give us after it has been massaged and sanitized".[9]

Only by using data that has not already been filtered and processed can the system help in problem finding and opportunity spotting. For an executive, finding problems and spotting opportunities may be more important than solving problems and exploiting opportunities. With appropriate information, top executives can spend more time motivating and aiming the efforts of line managers (the problem solvers) than doing problem solving themselves.

Analytical Capacity

Giving an executive access to detailed data is not enough. The EIS should have the capability to convert the mass of available data into *actionable information.* This entails, first, *access* to the data through the use of relational database management systems and well-designed menus that enable the user to home in on the desired level of data. Second, the system should avoid the *data overload* problem that is present in much of the computer output generated by traditional MIS systems. A screen-oriented system can act as a brake in this regard. Third, the use of *graphics* and *color* can contribute to the data conversion process. Trends can be seen in seconds as opposed to minutes or hours required to scan data in tabular form.

Finally, the system should not be limited to data retrieval and status reporting but should include some minimal analysis capability to evaluate the implications if the status quo continues into the future. The system should enable the user to *extrapolate* and *explore trends* to determine *when to intervene.*

Ease of Use

Three reasons for lack of enthusiasm for executive information systems from senior executives are: 1) their lack of computer training and experience; 2) their unwillingness to spend time getting such training; and 3) their poor keyboard skills. Our definition of "user-friendly" for an EIS deals with all three of these problems. Executives should be able to begin using an EIS without *any* prior training with the system. A simple demonstration or run-through should suffice to get any executive started on the system.

The dialogue between the executive and the system should *invite inquiry* and entice the user to *explore further.* The paths of inquiry should be optional and flexible but clear, and they should permit easy navigation through the database. Thus, the command language should be natural, transparent, and easy to operate and follow. A mouse and such modern touches as pull-down menus can help, but these are not essential. The key is a concise, straightforward system without the kinds of complexities that require voluminous user manuals and documentation.

Data Problems

The value of any information system hinges on the quality of its data. Often the data are inaccurate, incomplete, out-of-date, or inappropriate and irrelevant. Technically oriented system developers do not, in our view, give this issue the attention it deserves. More often than not they believe that data issues are outside the domain of their responsibility.

A common pitfall of existing databases in many organizations is that they were designed for accounting-oriented transaction processing applications and lack the detail required for *management* of information. For example, the order entry and invoicing systems in most companies do not include sufficient information on customers to permit strategic analysis of sales by user-defined market segments.

Another problem in building an EIS database is the difficulty of integrating internal data residing in different, incompatible computer systems. Apart from this problem, a more serious issue is that the EIS system has to look beyond internal data to meet the needs of top management. The existing databases contain primarily "hard" data or "the facts." What is missing is the "soft" data, such as information about industry trends and competitors' activities, which have to be searched out and put together for inclusion in the system.

A controversial question that must also be addressed in building the EIS database concerns the ownership of the data. Providing common access to operational performance information in a decentralized organization can be unsettling.[7]

We turn now to a description of a system developed for an agency of the New York state government that addresses these issues and gives the senior administrator in the agency up-to-date, easy-to-manage information in a format that is both powerful and useful.

The OGS System

The Case Environment

The EIS system described here was developed for the New York State Office of General Services

(OGS). OGS is a "conglomerate" organization having major operating units that provide a diverse range of services to other departments of the New York state government. Examples of the services that OGS provides include:

- Design and construction of state facilities
- Operation and maintenance of state buildings
- Wholesale food and laundry services to New York state correctional and health-related institutions
- Statewide vehicle management
- Statewide telecommunication and centralized data processing
- Office space leasing and land reutilization
- Centralized commodity contracting and standard testing
- Centralized printing

OGS has an annual budget of over $530 million and a work force of more than 4,200. Prior to 1986, the corporate culture of the organization could be characterized as passive. Annual budget increases were based largely on inflationary increases, and deficits incurred due to overspending were balanced by supplemental budget requests late in the fiscal year.

The situation changed in 1986 with the arrival of a new administrative head who stated the OGS mission as providing *efficient*, *effective*, *responsive*, and *economical* programs. A more formal management process for planning and control was instituted soon thereafter. The Human Resources and Fiscal Control System (HRFCS) was developed to implement the new process. The purpose of HRFCS was to decentralize the management and accountability of resources to the lowest practical program unit. Managers were required to budget for their needs and manage their expenditures.

As HRFCS was implemented, it became clear that "problem" areas, e.g., units that exceeded their resource allocation, could be identified only after the fact. Any solution had to keep track of expenditures as they occurred and extrapolate them through the end of the fiscal year to identify excessive spending rates.

It quickly became obvious that such capability did not exist in the agency's information system and that, to implement desired changes, the new administrator needed more powerful tools to track and analyze expenditure patterns—the kind of information that an executive information system might provide. The purpose of the system would be to provide the foundation for analysis and communication that would enable the administrator to signal to the organization that budgetary, planning, and control procedures had to change. A primary objective was to avoid a perennial multimillion dollar deficit.

Beyond expenditure control, the new administrator stated, "I want to know how we are performing against standards, not just against the budget." Clearly the determination of productivity measures and standards is a difficult problem, given the diversity of the operating units included within OGS, but it is also an important area for EIS development.

System Development

Development of the OGS EIS has followed an evolutionary plan, starting with an expenditure control module that was fully implemented in June 1988. The objective of this first phase of the OGS EIS was to develop an easy-to-use "drill down" system to show the current status of each program budget vs. expenditures and estimate expenditures through the end of the fiscal year to test the actual spending rate versus the budget. Subsequent phases will focus on the development of a performance analysis system to analyze and improve organizational productivity and effectiveness.

The OGS EIS has been implemented on a personal computer that is networked to OGS's mainframe computer for downloading of data. Currently, the downloading is done monthly. The system is interactive, screen oriented, and relies heavily on graphic displays of data. A wide variety of tabular and graphic reports are available. Printed reports and graphs can be produced on request right at the user's workstation. The key features of the OGS system development process are summarized below:

- A *quick* understanding of the administrator's information needs in terms of *what* he wanted and how he wanted to see it.

To show the administrator what the new technology had to offer, an EIS that had been developed for another public sector agency was demonstrated. The demo system had a different focus but had the screen-orientation and push-button graphical analysis features that the OGS administrator found appealing. The demo essentially settled the *how* question.

For the *what* question, the administrator was able to specify the broad objectives of the system in an initial two-hour project meeting. A second meeting was devoted to reviewing the formats of the reports and graphs produced by the system designer. At a third meeting a rough prototype with limited data was shown to the administrator and led to agreement on his information requirements.

In our view, the parsimonious use of the administrator's time in defining the system specifications was a critical success factor in maintaining his interest and involvement in the project.

- A relatively low-cost system.

As noted earlier, the difficulty of selling high-cost projects with no immediate, tangible benefits severely inhibits EIS development in the public sector, and in some private sector firms as well. The OGS system was implemented using standard, inexpensive PC software tools. The total cost of the system is about "two zeros" less than many private sector systems that have been implemented using specialized EIS software.

- "Push-button" operating features.

Dialogue with the system has been designed so that the user "pushes" only the cursor keys, the space bar, the zero (0) key, and the return key. The up and down arrows allow selection of menu items, moving the highlight from one item to the next. The return key selects the highlighted item. The zero key is always used to exit a menu.

The screens are carefully organized to expedite the selection process. For example, the initial highlighted item for the question "Is this correct?" is "Yes," the most likely response, requiring only a press of the return key. More sophisticated users may select menu items by pushing the alphabetic key corresponding to the item label.

- Ease of use.

The well-organized menus and a straightforward command structure combine to minimize confusion and make the system simple to use and understand. *No* user manual is provided or required. A 30-minute demonstration has proved sufficient to introduce uninitiated users to all system features and options.

The Data Cube

The database of the OGS system does not contain aggregated data. Rather, the user has access to a "cube" of detailed information, which can be "cut' in any of several dimensions and aggregated at any desired level (see Figure 1). For example, an expense category can be investigated for a given month across all organizational units, or for several months for an individual unit.

In addition to a large number of expense categories (e.g., overtime, travel), the New York State budget contains at least 30 different "appropriations" (the term used in the New York state budget for funding streams from different sources), which have to be managed separately. For example, some OGS employees are paid from state funds, some from federal funds, and some from other state agency budgets. The system keeps track of all appropriations and has easy-to-use screens for selecting the desired expense and appropriation categories.

The system also provides access to data at several different levels of the organization, down to

Figure 1.
The OGS Data Cube

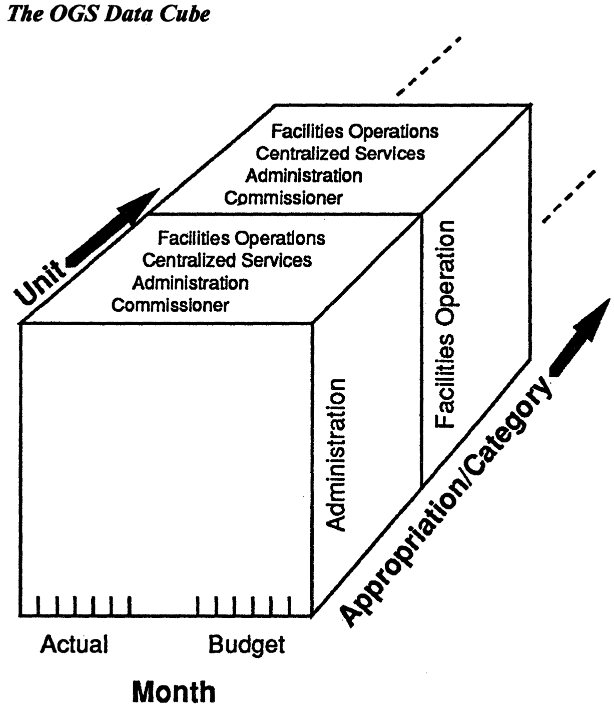
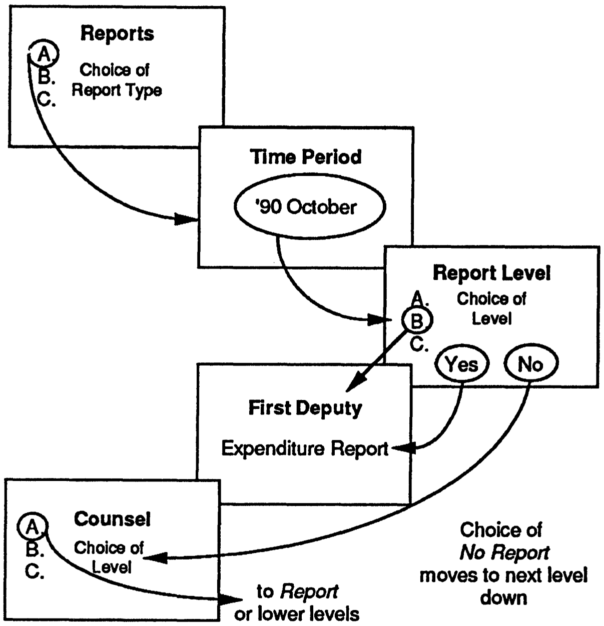


Figure 2.
Flow of a Typical Dialogue with the OGS System



the lowest level of control. An example of a "high level" would be the commissioner/first deputy commissioner level. A "low level" example might be the staff at the Binghamton State Office Building, who are within the Building Operations and Maintenance unit, which reports to the director of Facilities Operations and Maintenance, who in turn reports to the deputy commissioner of Facilities Operations, who is directly under the first deputy commissioner.

With all levels in the data cube available for access, the administrator can cut down through the layers of the organization to "follow" data to its source (or "drill down," to use the term in the Duracell example). For issues where higher-level aggregation is sufficient, the data can be reviewed at that level. In the previous example of levels, the administrator could look at the data for the entire Facilities Operations and Maintenance unit in which the data for the Binghamton State Office Building are aggregated along with data from all other facilities. The assumption is that the user will "drill down" only when more detail is needed to determine the source of problems identified at higher levels.

Working with The System

A diagram outlining the flow of a typical dialogue with the system is shown in Figure 2. Details of the screens for the same dialogue are shown in Figure 3. In this example the user has selected an *expenditure analysis* for a particular unit (item A on Screen 1). The system then prompts the user to select the time period (Screen 2). In the example it was chosen as October 1990. The next step is to select the organizational level of the desired unit (Screen 3). Item B, the first deputy commissioner level, was selected. At the bottom of this screen is a question, "Do you want report for this unit?" A "yes" response displays the screen with the report for the first deputy level (Screen 4). A "no" response leads into another selection screen containing organizational levels below the first deputy (Screen 5). From this screen, the user could pick a unit at this level or choose to drill down even further.

The number of organizational levels in OGS ranges from a minimum of three to a maximum of six, with the commissioner and first deputy considered as the first level. The complete structure of the organization is built into the system so that users are told when no further drilling is possible (as in Screen 6 of the example in Figure 3).

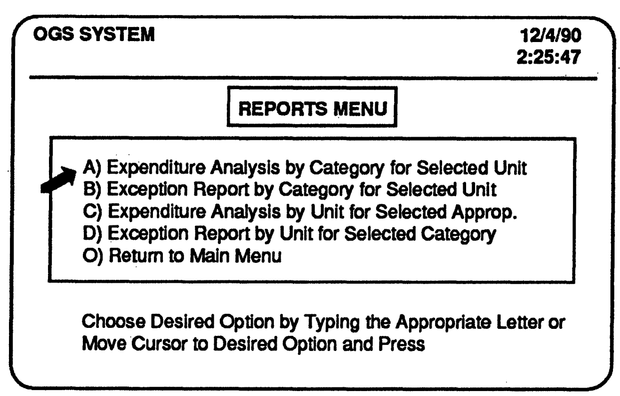
Analytical Features

The system is not limited to factual reporting of historical data. Several features are readily available to perform the following types of analyses:

Identify Problem Areas

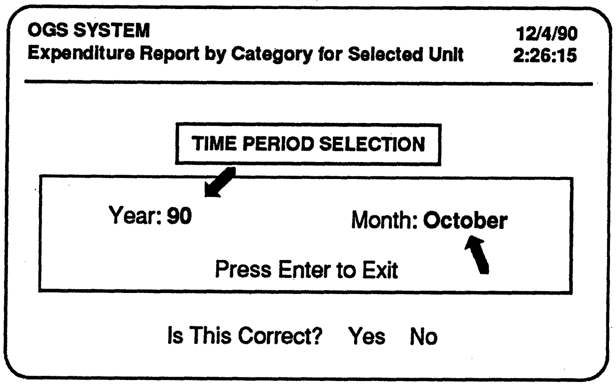
The exception reporting capabilities built into the system can spotlight either the appropriations and

Figure 3a.
A Dialogue Example: Screen 1



Screen 1

Figure 3b.
Screen 2



Screen 2

categories of expenditure that are over budget for a selected unit, or the units that are over budget for a selected appropriation or expenditure category.

Focus on the Specific Sources of Problems
The system design allows the user to cut through the complex organizational levels and categories to track down the real sources of problems. This cutting (or "drilling") through the data cube is one of the most important features of the system.

Evaluate the Performance of Individual Units
Graphics features are provided that permit the tracking of budget and actual spending patterns. Bar graphs can be used to compare the performance of several different units in achieving budgetary control. Cumulative trend lines can assist in determining whether an overspent unit is making progress to return to alignment with the budget.

Facilitate Early Detection of Problems
The system contains extrapolation and trend analysis features to provide an early signal of potential overspending. For example, based on the first three months of travel expenditures in a fiscal year, projections for annual spending can be developed based on historical spending patterns. Armed with such information, intervention can be planned at several different levels of the organization before expenditures go over budget. This helps to avoid future deficits and, even more important, provides evidence of administrative concern and resolve to affect change.

Expansion to Planning Activities
With the implementation of the expenditure control module, the focus of the project shifted to the planning function. In a people-intensive organization such as the OGS, which must also cope with the endemic budgetary constraints of the public sector, a key planning issue is: How many people are needed to perform the required work in each unit? This raises the following subsidiary questions:

1. What is the *workload* of the unit?
2. How is that workload *measured*?
3. What is the *performance standard* for accomplishing a unit of work?

Questions 2 and 3 bring immediately to the surface the problems of productivity measurement in large, bureaucratized public sector organizations and, for that matter, in many organizations in the private sector as well. To set the measurement process in motion, the OGS administrator has adopted an iterative approach focused on individual operating units.

For example, the unit that processes invoices started with no hard data on either workload or productivity standards. In attempting to estimate the time required to process an invoice, it immediately became clear that all invoices are not alike—the processing time varies with the type of invoice. This implied that workload data was needed not only on the total number of invoices processed but also broken down by the type of invoice. As a first cut, the invoices were classified into two major

Figure 3c.
Screen 3

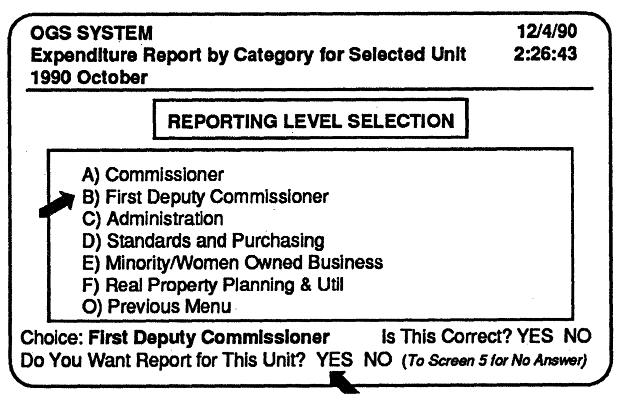


Figure 3d.
Screen 4

OGS SYSTEM UNIT: First Deputy Commissioner 12/4/90
Expenditure Report by Category for Selected Unit 2:27:24
1990 October

Appropriation/ Category	Budget To-Date	Exp. TDate +Encum.	Var.	Proj. Annual Exp.	Annual Appr. Budget	L.Y. Annual Exp.
Administration-SP/						
PS Regular	86700	99319	-12619	170261	148900	148900
PS Regular Overtime	425	0	425	0	732	734
Total PS Regular	87125	99319	-12194	170261	149632	149634
PS Temporary	1742	0	1742	0	3000	3000
Total PS Temporary	1742	0	1742	0	3000	3000
Total PS	88867	99319	-10452	170261	152632	152634
Misc Supplies	2303	1538	765	2637	3960	3960

Variance = Budget To-Date - (Exp. To-Date + Encumbered)

PgDn Next Page End Exit

Screen 4

types. Managers in the unit used their experience and judgment to provide processing time standards for each type of invoice. In many cases, these standards led to a headcount requirement that was about equal to the present staff complement. Nevertheless, it was a beginning to measure something that had never been measured before. Further, it provided a basis for tracking *actual* productivity against the estimated standards.

The iterative process under way will permit refinement of performance standards and enable more accurate analysis of fluctuations in actual workload. In this context, it must be pointed out that workload measurement and standard setting is a difficult task in some OGS units. In the training unit of the human resources department, for example, what is the appropriate measure of workload? This unit looks very much like a university faculty or a training unit in a large company, where workload measurement is also a very difficult task.

The second phase of this project has been completed in prototype form with detailed data from two operating units; this phase has been reviewed by the OGS administrator. He has since determined that the formidable task of developing data for all units will be worthwhile. The data can be converted into useful information by the EIS system to assist in planning and effecting productivity improvements. The prototype was a critically important step in building the administrator's confidence that the information produced by the EIS system would justify the significant effort needed to overcome the natural organizational resistance to workload measurement and standards development.

The productivity analysis module will present the administrator with a multilevel data cube similar to the one in the expenditure control system. He will be able to evaluate workload and productivity at the level where the work is performed and then institute the planning process from the grassroots level on a more factual basis. This system would provide him for the first time the opportunity to make high-level budgetary and resource allocation decisions based on a clear understanding of present and future workloads and productivity standards.

Conclusions

A primary observation on the implementation and use of the system is that the senior OGS administrator was absolutely critical to the project's success. Given the nature of public sector organizations, the importance of the commitment from the executive sponsor cannot be overstated. In the private sector, once the chief executive wants an EIS, it will move. In the public sector, wanting is not enough. Movement can stop at any of a number of stages. The OGS system is where it is today because the administrator moved it himself. A tangible result of his efforts to change the management processes in OGS is that fiscal 1988 was the first year in several years that was completed without a deficit. The second phase of the project is now moving beyond the prototype stage

Figure 3e.
Screen 5

OGS SYSTEM 12/4/90
Expenditure Report by Category for Selected Unit 2:27:43
1990 October

REPORTING LEVEL SELECTION

A) Counsel
B) Inspector General
C) Affirmative Action
D) OSHA
O) Previous Menu

Choice: **Counsel** Is This Correct? YES NO

Do You Want Report for This Unit? YES NO *(To Screen 6 for NO Answer)*

Screen 5

Figure 3f.
Screen 6

OGS SYSTEM 12/4/90
Expenditure Report by Category for Selected Unit 2:28:53
1990 October

No Units at the Next Level
Press Any Key to Choose Other Unit

Screen 6

and is targeted for completion in time to use it for planning the 1991-92 fiscal year.

Development of the database to support the EIS has not been without problems. The downloading of data from the OGS mainframe computer to the executive workstation has posed the usual problems of compatibility, data formats, discrepancies, and gaps in the data. More importantly, for the second phase of the project, much of the data on workload and standards does not exist. The prototype that has just been completed for this phase has proven useful in resolving several issues relating to the collection and processing of the required data.

In conclusion, our experience highlights the importance of the following factors in successful implementation of an EIS in the public sector:

1. The role and commitment of top management
2. Modularity in system development
3. Prototyping with live data
4. Push-button ease of use
5. Low cost
6. The need for detailed data in the system
7. The ability to access data at any level, detailed or aggregated
8. An iterative approach in developing new data where none existed

References

[1]Brody, H. "The Executive PC: A New Light on Business," *PC/ Computing*, December 1988, pp. 152-157.
[2]Rockart, J.F. and Treacy, M.E. "The CEO Goes On-Line," *Harvard Business Review* (60:1), January-February 1982, pp. 32-88.
[3]Kiechel, W., III. "Why Executives Don't Compute," *Fortune*, November 14, 1983, pp. 241-246.
[4]Rockart, J.F. and Treacy, M.E. "Executive Information Support Systems," working paper, CISR No. 65, Sloan WP No. 1167-80, Sloan School of Management, Massachusetts Institute of Technology, Cambridge, MA, 1980.
[5]Main, J. "At Last, Software CEO's Can Use," *Fortune*, March 13, 1989, pp. 77-83.
[6]Houdeshel, G. and Watson, H.J. "The Management Information and Decision Support (MIDS) System at Lockheed-Georgia," *MIS Quarterly* (11:1), March 1987, pp. 127-140.
[7]Rockart, J.F. and DeLong, D.W. *Executive Support Systems*, Dow Jones Irwin, Homewood, IL, 1988.
[8]*Time.* "The Granite State of the Art," January 27, 1986, p. 70.
[9]Beeby, R.H. "How to Crunch a Bunch of Figures," Wall Street Journal, June 11, 1990, p. A14. ■

Downsizing to Microcomputer LANs

In this report:

Datapro Summary

Downsizing is a process that is being executed by many cost-conscious companies. But beyond the cost issue, downsizing is an efficient way to free the corporate mainframe from processing that can be effectively handled by smaller, capable systems. If not approached in the correct manner, the downsizing process can cost more than it is worth. With proper planning, downsizing corporate computer facilities can prove beneficial for all involved.

Eventually, even the most die-hard fans of the mainframe may have to consider the possibilities of moving to PC LANs. While the idea sounds attractive—especially in terms of savings in capital expenditure, overhead, and those big salaries that you will save by firing the mainframe whiz kids—the move from mainframe to PC LAN may not go so smoothly. The savings may soon disappear and you may find yourself recruiting more support staff to help more users.

Downsizing must be attempted with prudence and planning. It can be rewarding if the expectations are realistic and the move is carefully planned.

"It's not easy to anticipate all problems involved in downsizing," observes Anthony Townsend, microcomputer support coordinator, University of Virginia at Charlottesville.

When the University of Virginia recently downsized many of its mainframe-based applications, it faced some unexpected problems. Some vendors changed their policies and their applications ended up costing more. Also, management agonized whether it should buy off-the-shelf applications or develop custom software. Townsend warns potential downsizing enthusiasts that it's a long, drawn-out process that can deliver some shocks to those attempting it.

What is Downsizing Anyway?

The word "downsizing" is often misused. Downsizing evokes visions of drastic reduction, layoffs, lost opportunities, and for some, an opportunity to make a few bucks. The industry is still struggling to come up with a more acceptable and less confusing term to describe the process of shifting appropriate processing from mainframes to PCs and workstations.

What is the main purpose of downsizing? Is it to make MIS lean and mean? Is it to make it easy to manage the computing facilities? Is it to give a better deal to the users by providing easy access to data? Is it to save a few bucks? Or is the main purpose of downsizing to make management of the data processing facilities more efficient?

Organizations' management often seem to be unaware that downsizing's real objective is to free mainframes from processing data that can be done by compact and inexpensive machines and that its purpose is not merely a cost-cutting opportunity.

This Datapro report is a reprint of "The Up Side of Downsizing" by Anand Rao, pp. 51-60, from *LAN Magazine*, Volume 6, Number 6, June 1991.

The existing MIS structure evolved from an "us and them" attitude between the users and the managers of computing facilities.

As computers grew user friendly and the power of PCs and workstations increased, the iron door between the users and the computers began to rust. Unfortunately, the same can't be said for attitudes. The user and MIS communities are gradually becoming aware that concentrating most data processing on mainframes is grossly inefficient and that a catastrophe won't occur if users are given more access to computing facilities. With greater access, it becomes more efficient to move the facilities closer to the users. The compact computing resources located closer to the users are easy to install, maintain, and use.

The process of downsizing doesn't mean decreasing the type, magnitude, quality, or variety of processing facilities that are available to users. Downsizing can actually increase the overall quality and quantity of services available. Innovations in networking and desktop computer's processing power has made it possible to bring mainframe computing power to the user's desk.

Even the early proponents of downsizing soon realized that the process involves much more than getting rid of mainframes, firing some programmers and operations personnel, and waiting for the enormous savings to occur. The mainframe can't be totally replaced by desktop computers, and it's counter-productive to shift everything from a central location and distribute it over the facility.

The mainframe can be retained to perform number-crunching applications and transmit formatted data to the users via the network's communications facilities. Additionally, downsizing can be limited so it fits the facility's requirements.

When It's Time to Downsize

While it's easy to enumerate the benefits of downsizing, it's not so simple to decide whether it's appropriate to the facility's needs, whether all its promises can be met, to what extent it should be implemented, and how long it will take to downsize fully.

In practice, most data processing facilities are poorly managed and under-utilized. Many MIS managers who face the wrath of users every day wonder if a better way to process data and please the users exists.

In most organizations, MIS managers spend the first couple of hours of every day listening to a litany of user complaints: Reports unavailable, frequent downtime, too many maintenance interruptions, frequent program crashes, and data formatted to suit MIS, not the users.

As a rule of thumb, it's time to downsize when user complaints get out of hand, when too many users are waiting too long for the program to come up, when too many "log off, the application is about to crash" messages are flashed on users' screens, and too many routine processing functions are performed on the mainframe facility.

Other telltale signs of overload include erratic behavior of the system when it hits near maximum load, clogged backup procedures, too many problems associated with start-up and shutdown functions, and above all, far too many occurrences of facilities being unavailable because an expert staff isn't available.

A discernible pattern of the facility's shortcomings will materialize if you perform a thorough analysis of the problem log and user complaints over a three-month period.

The MIS management must decide if the problems can be eliminated with better operating procedures or if a radically different way of processing the data is in order. Essential to the decision is whether the users will accept the changes and whether the management will support the move.

What to Expect

Expecting too much while deciding to downsize is easy. You can avoid this trap by making a list of expectations and thoroughly analyzing them by taking into account the views of the company's management, users, and MIS staff.

Are the management's expectations purely motivated by bottom line benefits or is management prepared to trade off expenses for efficiency? Does management appreciate the whole process?

Present a report that details downsizing's process and benefits to management. The MIS people who will actually manage the downsizing must understand management's expectations and make sure that the people who write the checks clearly understand what is being performed.

The users who complain the loudest often seem to have the least understanding of MIS's problems. Users, usually represented by a user group, must understand what downsizing will achieve and what they can expect from it. A list of user expectations should be prepared and matched with a list of management's expectations. This procedure will help clear up some of the users' misunderstandings about downsizing.

The MIS staff, a critical component, is often taken for granted, even when concerning important decisions that change the system's structure. The MIS staff should be consulted; and its views, fears, and expectations should be recorded.

Following this process will help ensure that the three vital components of the move understand what is involved, what to expect, and most importantly, what *not* to expect from downsizing. Use the information gathered in this process to prepare a list of primary forces that motivate those concerned to accept or reject the move to downsize.

Next, classify the driving forces into prime issues such as bottom-line expectations, user ease, data integrity, and operational problems. Downsizing is not merely a technical operation that is transparent to the users. Downsizing brings computing closer to the user community, thereby forcing a decisive change in the way people in the company work. Everyone must know the consequences of the change and how it will affect their work habits.

Planning to Downsize

Although detailed planning is perhaps the most important issue, it is often performed halfheartedly—and with disastrous consequences. Downsizing can't be performed incrementally. All vital components of the move must be planned.

One of the biggest impediments to planning is the lack of reliable data. Often, the volumes of available data from the mainframe can't be used on the PC LAN because the file and data formats are incompatible. You must generate meaningful reports from the available data so you can find out what is not available currently. Identify the information domains and the appropriate reliable sources of information for each.

Perform a detailed inventory of the entire facility, including all hardware, software, facilities, operational statistics, and dependency schematics. While most MIS shops maintain a fairly up-to-date inventory, the focus of taking inventory for the purpose of planning a downsizing is different.

Perform a complete inventory of hardware in the MIS facility, tape room, and user areas.

Make a complete list of software used in the systems area along with the applications used by the users. Maintain a separate list of software that was coded in-house and bought off-the-shelf.

Take a complete inventory of all facilities, including existing ducts, wiring schemes, vertical risers, and patch panels, and prepare an appropriate map.

Prepare operational statistics on the usage of applications, including the number of times and how long each application is used. Also prepare an identification of the users. A pattern of usage study should be developed and all rarely-used applications should be listed separately.

Prepare a graphical representation of all application dependencies. Most shops don't have dependency schematics, which keeps the MIS staff guessing as to what would happen if one of the applications is made redundant. The schematic should also include user access graphics that depict who is accessing what through which facility, how many times per day, and for how long.

Collecting reliable data can be trying; indeed, it can be the most difficult part of the operation. Identifying reliable sources of information is not easy. To generate a consistent format for statistics, you should develop questionnaires and test them before generating live data.

After collecting the initial data, match it with the primary driving forces that influence expectations of each segment of the organization. At this stage, planners should prepare for surprises. Some expectations may prove unrealistic, while others can be met by making minor changes to the facility in its current state.

The data should be used to develop goals for the operation. The goals should include, short-, medium-, and long-term project milestones.

This is the time to think about the effect that downsizing would have on the corporate culture. It may surprise MIS, management, and the users when they realize the amount of waste and redundancy that occurs in the facility.

Nine Steps to Success

How to Make Downsizing Work for You
Most downsizing projects either fail or incur extreme cost overruns and delays because of improper planning and management. Projects spin out of control because problems aren't solved in the beginning and then snowball into unmanageable proportions. Here are nine keys to successful downsizing.

1. Document Management
Carefully assemble and organize all management reports, vendor agreements, schematics, and project management reports.

2. Facilities Management
Pay close attention to procedures of equipment removal, equipment receiving, and wiring.

3. File Conversion
A responsible person must coordinate the conversion of mainframe file formats to PC LAN file formats.

4. Transition Management
Ensuring the continued availability of services during the transition is tricky. Examine facility closures right from the start.

5. Service Agreements and User Groups Issues
Keeping all users apprised of the changes will allay their fears.

6. Testing
First try downsizing as a pilot program, then continue to test after the downsizing.

7. Training and Support Services
Formalize training for users and formulate your plan for continued support services.

8. MIS Staff Management
Address the retraining and redeployment of the existing MIS staff.

9. Backup and Security
Define new backup, security, and disaster prevention methods and procedures that will fit the radically altered environment.

Operation Downsizing

When all the data is collected and meaningful reports are generated, the project managers will face some of the most difficult decisions of "Operation Downsizing."

What to Keep and What to Throw Away?

This depends on the project's overall goals and the extent of downsizing. You can assume that each piece of hardware and software currently installed performs a useful role. The planners must analyze issues such as: Can we do without it? Can it be replaced by a better technology in the downsized environment? What will it cost to replace the item?

For each piece of mainframe hardware that can be junked, the planners must consider the software-related issues. Can the data be used in the downsized environment as is, or is extensive file conversion required?

How Long Will It Take?

Data processing and networking projects are notorious for overshooting budgets and time limits. Some of the most difficult issues to estimate in downsizing relate to estimating the time required for rewiring, testing, installing the appropriate boards in all user facilities; installing file servers; and file conversion.

A project often takes three times longer to complete than the original estimate. While you can make ballpark guesses by consulting with the appropriate vendors and subcontractors, it's more practical to make estimates by performing some elements of the project on a test basis.

What Is the Realistic Transition Time?

Even after the project is completed, it usually takes some time for a complete turnover. Quite often, conversion projects languish in the transition stage for long periods while additional changes are made to the project itself. While planning the transition, the planners should consider the stages of downsizing, what will be downsized first, how long the facility will have to function in a parallel mode, and what are the appropriate accommodations that need to be made.

What Are the Personnel Requirements?

The expertise required for the project usually revolve around operations, converting file formats, pulling wire, installing equipment testing, and training. While vendor-supplied personnel can perform some of these operations, it's also possible to use in-house personnel. Consider issues such as whether selected personnel can be spared from their current duties, how many on the current staff will need training, and how many will be let go. Personnel requirements for each project segment, skills requirements, availability, and duration should be considered.

What Additional Hardware and Software Are Needed?

List the additional hardware and software required in the user and operations areas. Vendors can be helpful, but they must receive detailed plans to make accurate estimates. Since these estimates depend on the number of users, LAN configurations to be installed, and file servers, you must determine the extent of downsizing while planning this segment.

Also, organizers must decide how files will be converted, which software will be bought off-the-shelf, and which software will be custom-coded for the new environment.

Cable Management Challenges

Cable management can prove more difficult than originally planned. Prepare in advance a complete schematic of the facility as it will look when it's downsized. This demands a thorough survey of existing wiring facilities, outlets, patch panels, drop cables, and concentrators, as well as the wiring schemes to be implemented in the new LANs. While some people consider both logical and physical connections at this stage, it's prudent to determine just the physical wiring schemes.

Consider issues such as whether to use existing phone ducts or build new ones. Depending on the types of cable to be used, the hazardous areas that the cable will pass through, and the problems of electromagnetic and other interference should be considered.

Wiring also depends on capacity planning, future growth projections, the installation of a backbone LAN or a super-backbone facility.

Project Costing

While project costing methodologies differ according to company culture, a consensus on some issues can be reached. Project costing should be related to: capital expenses, wiring, installation, training and user support, and operational and maintenance costs.

1. *Capital Expenses:* What new hardware and software will be required in the downsized facility? Since capital expense is likely to increase incrementally with the number of users, type of applications used, and the growth in the overall size of the facility, you should consider the costs of accomplishing each stage of the operation.
2. *Wiring and Related Expenditures:* Wiring expenditures depend on how it will be performed. Most vendors quote prices per project that include both installation and material costs. Since the vendors' prices depend on the amount of labor involved (often as high or higher than the cost of the actual cables), it is essential that you have made a thorough survey of the existing facilities. Although it's not uncommon to have to rewire to avoid disturbing existing facilities or reconstruction, rewiring can be avoided with the proper planning.
3. *Installation Costs:* Installing adapter boards, file servers, and software, and testing it all can be expensive. It's better that the in-house staff perform this job, since it's difficult to estimate the time required to perform all installations. Plus, some installation will be ongoing.
4. *Training and User Support Charges:* Training the users and the operations staff involves a considerable expenditure. Users will require on-the-spot training and in certain cases, they will need some classroom instruction. The current MIS staff will require additional training to function in a LAN shop. The in-house staff can also conduct training on an ongoing basis to cut costs.
5. *Operational and Maintenance Costs:* These personnel-intensive costs should be considered in advance; however, they're difficult to determine accurately.

Avoiding Cost Overruns

Few data processing and networking projects have ever been carried out without cost overruns. While cost overruns can be minimized, don't expect to eliminate them altogether. Better planning and preparations can eliminate some chronic project costing problems. Cost overruns come as nasty surprises when work is started without a complete survey of the existing facilities, the work required, and how the project will look when completed. The only way to avoid cost overruns is to plan ahead and draw on the experiences of other organizations that have already downsized.

One crucial element in avoiding cost overruns is effective vendor negotiations. Vendors should be consulted to minimize cost overruns and you should enter into written agreements that cover all aspects of the project before the work starts. However, even experienced project managers will admit that cost overruns are a part of managing a project.

Managing the Downsizing

Even the best planned projects have a way of spinning out of control unless they are managed effectively. Managing a downsizing project involves issues that relate to migrating from one environment to the other.

Most projects of this type either fail or incur extreme cost overruns and delays because of improper planning and management. Usually, problems that go unsolved in the beginning snowball into unmanageable proportions later on.

Adopting some features of network management protocols and developing some specific techniques that are unique to downsizing can help manage downsizing projects.

1. *Documentation Management:* Many projects fail because the documentation is poorly managed. The project manager must have all the documents on hand and institute a plan to make them available as needed. Additionally, documents must be properly updated, numbered, and renumbered. The document library should be well organized. Appoint a document manager, preferably a person who has experience in document management.

   While the type of documents required are specific to each project, some are common to all downsizing projects, including management reports, vendor agreements, schematics, and project management reports.

   Prepare a well-documented management report that details the downsizing issues, including the project's scope, cost, time required, benefits, and other factors that may influence the project. Then obtain management's approval for the report. All agreements with the vendors must be formalized and all issues must be fully clarified before the job orders are issued.

   All schematics related to the projects must be prepared after a thorough survey of the facility. These schematics should include graphics relating to the facility, wiring, locations of hazards, patch panels, concentrators, locations of the existing equipment, and configurations of the downsized facility.

   A project management report that details the project's various stages, important milestones, costing signals, reporting procedures, and assignments should be prepared in advance and distributed to important people associated with the project.

2. *Facilities Management:* Pay special attention to managing the facilities, which include equipment removal, equipment receiving, and wiring. Subdivide this into manageable components and assign management responsibilities to an appropriate person.

   Designate a person to oversee the removal of the existing equipment. Give him or her a list of what will be removed. This person should be responsible for getting a sign-off for each piece of equipment removed. Examine the facility in advance and remove obstacles, if any, that might hinder equipment removal.

   You must manage all aspects of receiving and storing new equipment. Provide the person responsible with the necessary documents to ensure that the new equipment conforms to the required standards.

   Wiring is an important part of the project. The person overseeing wiring should have appropriate qualifications. Supply him or her with the necessary schematics and reports. Projects are often delayed because of wiring mistakes.

3. *File Conversion Issues:* File conversion, an important component of downsizing, should be managed by a responsible, experienced person. Issues of file conversion include the existing file type and format, the type of conversion they will undergo, the procedures to be followed, know-how required for the process, and the duration of the conversion.

4. *Transition Management:* Managing the transition and ensuring the continued availability of all or some of the services during transition can prove tricky and difficult to control. Surprises await even well-prepared managers. Transition management requires that you address issues, such as outsourcing some processing facilities, changing access methods and procedures, advising or training users, data integrity, and security. Examine at the outset decisions such as whether facilities will be closed for some period of the transition.

5. *Service Agreements and User Groups Issues:* Regularly consult users and apprise them of the changes taking place during downsizing. Hold regularly scheduled user group meetings to address their concerns. Additional user related issues such as training and support should be addressed at each meeting.

6. *Testing:* Perform regular tests during and after downsizing. Test methods and procedures should be well-documented. The method of reporting the results should be formalized.

7. *Training and Support Services:* You should formalize training for the users based on the information gathered during user-group meetings. Training can be held in classrooms or at the users' sites, as the users require. Decide your support services plan, including managing the help desk, training the help desk personnel, and the type of help desk tools to be acquired. Additionally, plan whether some training will be done by the vendors or outside contractors.

8. *MIS Staff Management:* Address issues of retraining and redeploying the existing MIS staff and whether new expertise should be hired. Address the type and duration of training, whether it can be done in-house or outsourced, testing the quality of the training, as well as defining career goals for the staff in the new environment.

9. *Backup and Security:* Since the shop's environment will change radically, address issues related to backup and security, including defining the methods and procedures, responsibilities, and disaster prevention.

Downsizing: Ups and Downs

Downsizing helps to get rid of the attention- and resource-guzzling mainframes and helps to make the data processing facility more compact. But it also enforces profound changes in the company culture and operating procedure.

The issues that are the most pressing and difficult to handle may not be technical at all. Instead, management will have to address the dominant fear factor and help many people who are affected by the move. Management will have to assure the MIS staff that they won't be let go (if that really is the case) and promise them job security and career goals in the downsized environment. To complicate matters, management may have its own reservations about the project's success. MIS must make the appropriate assurances to allay the fears of the management who worry about staying competitive in the downsized environment. Users may fear the transition, difficulties of using the facility, data ownership, and understanding how the system works.

For downsizing, the areas of projected savings include operational overheads in managing the mainframe environment, licensing fees, and maintenance, but the saving generated may be offset by expenses in acquiring new equipment, training, wiring, operational overheads, licensing fees, and maintenance. The benefits should be viewed more in the ease of using and managing the facility than in cost savings alone.

The University of Virginia's Townsend says it didn't save money by downsizing, but it ended up getting a better facility that is easy to manage and the users like.

As long as you don't expect miracles, understand that problems can't always be anticipated, know that the savings in throwing out the mainframe can be offset by other expenses, and know that it takes expert management skills to manage the project, downsizing is the way to go. ■

Case Study—The Evolution of a LAN

In this report:

Datapro Summary

Implementing a networking system that holds up to the heavy demands of database-intensive applictions is an evolutionary process. The trials and tribulations of LAN implementation as experienced by SBT, a developer of popular accounting software, are experienced by many organizations. Ten years after the company installed its first Corvus Omninet, SBT has, through trial and error, developed a state-of-the-art networking environment.

If you are concerned about the rigors of heavy database demands on a local area network, SBT is an instructive case.If you write PC applications, you probably know what SBT is all about. The company sells accounting software with source code for the various base (Dbase) products in the DOS, Xenix, VAX/VMS, and Mac environments. What you may *not* know is that SBT subjects its own software to intense network use. The tactics they've come up with to improve performance during a decade of pounding their LAN are tricks every person responsible for multiuser Xbase applications should know.

SBT may not hold the world record for nodes, but unlike most LANs, all 148 workstations at SBT's headquarters are banging away with disk I/Os all day (see Figure 1). Many LANs in business organizations are characterized by heavy use of such sedate applications as spreadsheets, word processors, and electronic mail. They tend to load once from the file server, and for the duration of the session users rely on local resources. As a result, data traffic over the network cable is sparse and the server's disk gets plenty of R&R.

SBT's local area network operations are far less sedate. Users are pushing server disks by posing FoxBase+ and FoxPro queries and posting update transactions. Backups of time-stamped transactions are made on an hourly basis.

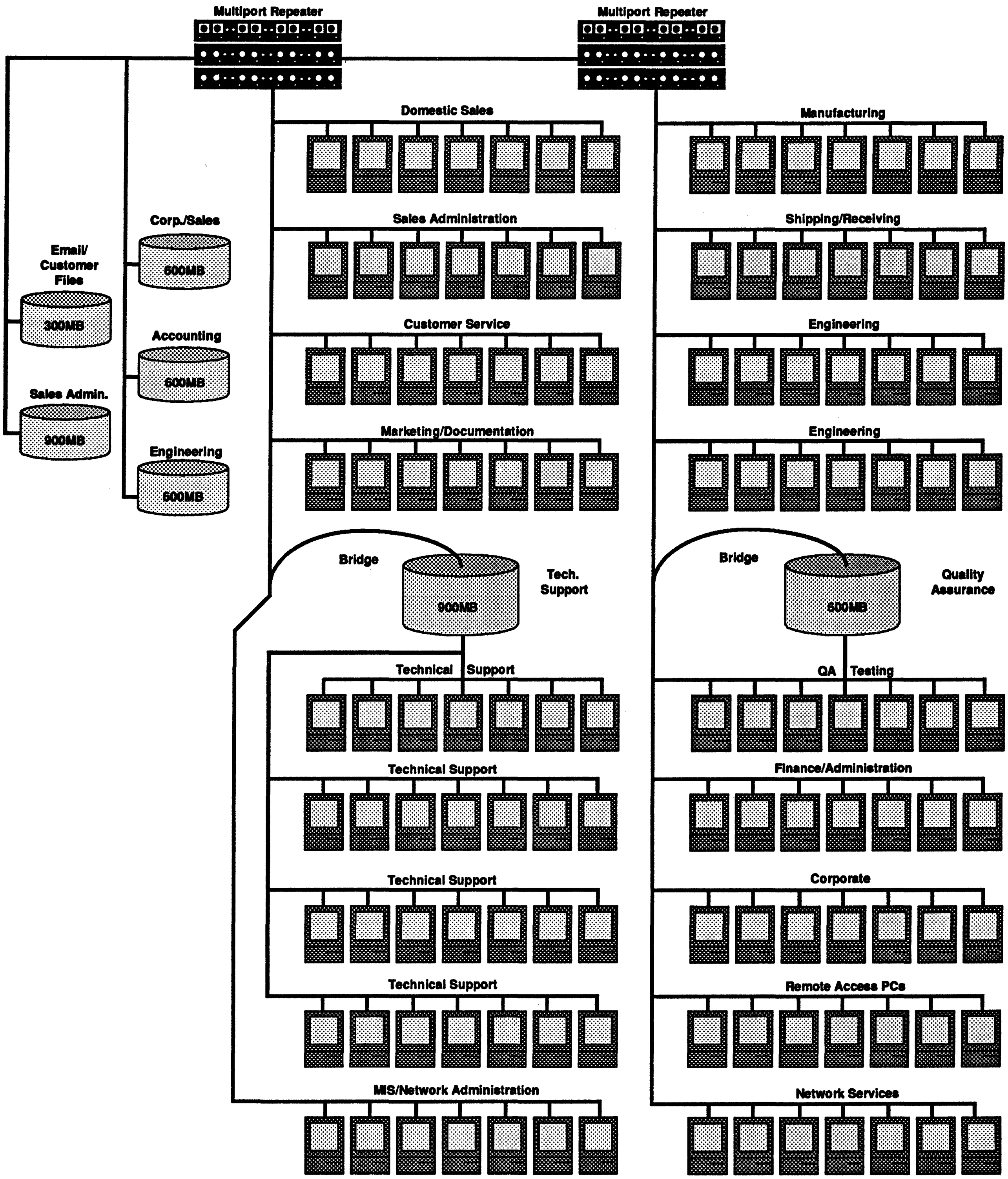
SBT doesn't just sell accounting software—the company uses it as actively as any of its customers. There are the obvious accounts payable and accounts receivable activities. Telemarketing personnel perform order entry to meet next day delivery. Marketing queries analyze sales trends. Monthly, weekly, and daily reports track the company's health. SBT President Robert Davies even likes to show visitors that he can track any department's performance over the last hour by running a real-time report.

Meanwhile, the development staff is thrashing the disks compiling code and the quality assurance team is banging away at the fruits of development.

Unlike spreadsheet jocks, database users are constantly reading and writing data stored on disks; SBT's users all seem to need access to a single file, the customer data file. (SBT developers will recognize AR Cust01.dbf as the the all important hub of activity and, potentially, a performance dampening bottleneck.)

This Datapro report is a reprint of "Network Know-How" by Marc Schnapp, pp. 38-46 from *DBMS*, Volume 4, Number 5, May 1991.

Figure 1.
SBT's Corporate Network



How SBT whipped its LAN into its current satisfactory shape requires a step back into the dim, dark ages of microcomputer LANs.

Ten Years Ago

SBT installed its first network in 1981 using Corvus Omninet. The system resided on CP/M and Apple hardware. The Corvus locking protocol required a more docile user base than you're likely to encounter today. Only one user at a time was granted write status—all others were granted read-only rights. Even with 25 employees, the system was cumbersome. SBT was plagued with cross-linked files and other networking gnats.

The release of multiuser dBASE III made a move to MS-DOS imperative. The system consisted of one server, 25 workstations, Ethernet, and Novell NetWare Version 1x. Novell offered greater performance and integrity than had Omninet.

It was clear after SBT's head count had doubled that multiuser dBASE III, though reliable, was inadequate for the load. The company snapped FoxBase into place, and suddenly things were moving as much as ten times faster. SBT, meanwhile, was benefiting from mushrooming demand for PC accounting software and was still growing. So was its customer data file.

By late 1988, Davies and his MIS director, Dave Harris, were becoming far too aware of the little red disk light on their server. The disk seemed to be thrashing; quick analysis revealed that the customer file was indeed the primary bottleneck. The MIS staff concluded that it was time to distribute disk I/Os by adding servers; that clinched development's realization that their accounts receivable module would be more effective on large LANs like theirs if they could distribute data files used by their software among multiple servers. But at the time Novell didn't offer multiple name services and 3Com did.

There were other lures to greener pastures. When 3Com's 3+ Share was taken down, one could apply standard DOS software like the Norton Utiliues to any glitches that might arise. The only way to fix some such problems under NetWare required the lengthy (24 hours or more on large disks) process of COMPSURFing server disks.

NetWare had another annoying problem that 3+ promised to. If a workstation went out of commission under 3+, one could bring it back up; any packets destined for the errant node when it went down would be redispatched automatically. This resulted in a huge operational difference, according to Davies. Under similar circumstances, the immature NetWare "trashed files."

SBT installed 3+ and threw two more servers at the problem. This allowed Quality Assurance to test multiple-server support for SBT's A/R code; the ever popular customer file was awarded a server of its own. The second most accessed data file, for inventory detail lines, was given similar treatment. Performance improved dramatically.

When operational benefits accrue to SBT, the company typically rolls these design features into its products. So SBT added multiple server support to its accounting software.

As the network approached 100 nodes, capacity was again challenged. SBT's current MIS Director, Dan deGrazia, cites frequently disruptive 3+ network retry errors—data packets were not reaching their destination. SBT's developers and quality assurance staff, by inclination the most sensitive to performance, were clamoring for a remedy. Harris frequently brought down the file server to readjust network RAM buffers and increase the number of file handles. The results were inconclusive. Exacerbating the pressure for relief was 3Com's insatiable RAM appetite. After loading the requisite device drivers, 3+ left just a bit more than 400K free on the workstations. Some within the company wanted to upgrade to FoxPro and needed at least 512K free. Technical support staffers needed 40K more to accommodate a TSR that let them hot-key into Xenix to field questions about SCO FoxBase versions of SBT.

Table 1. Performance Improvement

	Time in Minutes	
	NetWare 386	3Com3+
DetailInvoiceRegister	5	120
SummaryInvoiceRegister	4	45
Agedreceivables	10	45
CashReceipts	7	45
SalesTaxReport	5	180
SalesAnalysis	15	72
OpenReceivables	4	30
A/RJournal	20	240

A Solution

While 3+ was running out of steam for SBT, Novell was running away with the network market. NetWare deficiencies that had driven SBT away just a few years earlier seemed to have been ironed out by the vendor. And there was no question that SBT was going to have to make a move of some kind. The only question was how much of the bottleneck was related to 3Com's 3+ and how much to the network infrastructure?

Where once options had been limited, multiple alternatives were emerging. In addition to a move back to Novell, SBT considered Microsoft's OS/2-based LAN Manager network operating system. 3Com was very aggressively selling Harris on Microsoft's multimillion-dollar R&D commitment. "It's hard to ignore 3Com when they tell you they're in bed with Microsoft," Harris says.

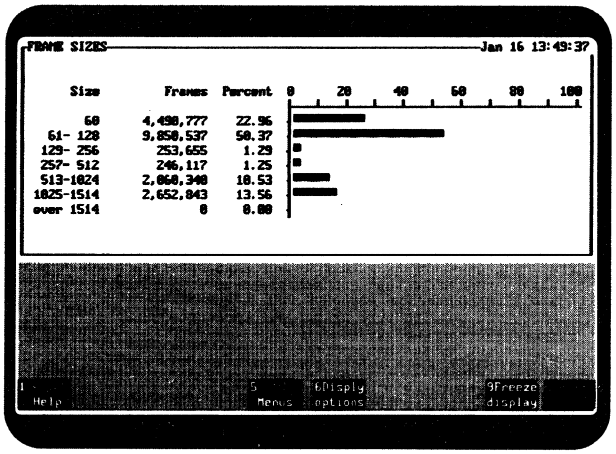
But when it came to the test, 3+Open (3Com's version of Manager) presented more questions than answers. During a pilot 3-node test, Harris had OS/2 setup problems, requiring him to switch to Compaq's version of the operating system. The server on the 3+Open test was prone to dead-halt crashes. "I felt like I was in the middle of a 3-ring circus with the operating system, the LAN software, and the 3Com extensions." For all the effort expended, Harris found 3+Open yielding a meager 20 percent speed improvement over DOS-based 3+.

deGrazia, however, feels SBT could have wrung better performance from 3+Open and eliminated the server problems: "The frustration and the learning curve were profound. It would have been equivalent to absorbing a minicomputer and a new operating system."

Novell's NetWare 2.15, in contrast, was a mature, known quantity. It was clear that many improvements had been made since the days when SBT had run its business on NetWare version 1 and dBASE III. NetWare installation was much improved. COMPSURF was much faster, and better yet, it was rarely needed—NetWare now applied online "hot patches" to bad disk sectors.

In a pilot test, then MIS Director Dave Harris installed NetWare 2.15 in the Quality Assurance area and applied merciless server disk read tests. Not only did Novell handle the abuse, it provided better performance during testing than the QA staff had experienced under more typical conditions. RAM crunch was no longer a problem, and the packet retry problem virtually disapeared. Any questions

Figure 2.
Watchdog at Work



about the adequacy of the 3Com topologv (Ethernet with thin cable and mostly 8-bit network interface cards) were wiped away.

The bottom line was that NetWare 2.15 was reliable and fast, significantly cutting processing times in a 5-node test. And there were rumbles that even more improvement could be had from NetWare 386, which was nearing release. Ordinarily, Harris's rule in life is don't go with version 1.0. "But after chatting with six key NetWare 386 beta testers, he was convinced that SBT should move to the more powerful version of Novell's network operating system.

There was another, quite compelling business-related reason for SBT to go with NetWare: MIS wasn't interested in wandering too far from the environment SBT customers were most likely to exploit, and Novell commanded about 70 percent of the LAN operating system market.

When SBT installed NetWare 386, gains in production applications were dramatic. End-of-month business reports ran 3.5 to 35 times faster than had 3Com's 3+ (see Table 1). Unfortunately, SBT can't directly compare NetWare 386 performance to that of LAN Manager-based 3+Open. Because installation and testing of both NetWare versions had run so smoothly, SBT decided not to pursue the OS/2 path further; comparable benchmarks were never run.

Fine Tuning

It was with the NetWare decision behind them, and installation still ahead, that SBT hired deGrazia for his experience with large NetWare networks (including a 450-node LAN) to lead them forward. Dave Harris, who had been in charge of the network, moved up to the post of VP of MIS and Quality Assurance.

deGrazia set out to study the particularities of server-based Xbase applications and how SBT relied on them.

"At least 40 percent of the people here are connected to the customer file most of the day," he says. The file weighs in at 127,000 records, requiring 78 MB of storage. While an Xbase query could pull large data streams down the network wire, the more typical use is to read and update individual records. That activity moves much smaller packets. On a typical LAN, packet sizes of 129-512 bytes comprise 60-70 percent of the traffic. Using The Network General Watchdog (Figure 2), Dan discovered the majority of data packets run between 60 and 128 bytes at SBT. Active networks with large numbers of small packets experience higher frequencies of packet collisions. This inevitably slows response time; Ethernet must re-transmit all colliding packets after pausing for random periods. Would this lead to trouble?

After installing NetWare 386, deGrazia applied a test imposing high transaction rates on the network. On 93 of the 125 workstations then on the company's network, he ran a looped record-skipping routine against the customer file. After skipping a couple of records, each workstation would display a record. "The pause on each station was never more than a second between displays," says deGrazia. "And we never dropped a single station because of network errors. SBT's network nodes are now approaching 150 with no significant degradation.

It seems SBT has made a sound choice.

SBT's Hardware

Through all of the changes, SBT has been largely faithful to its hardware.

Of course, there's a strong trend toward upgrades to 386-based machines as workstations and 486 machines as new servers. Improved disks are in evidence as well. The capacities of the most of the drives that deGrazia installs are going up to 600 MB. He seeks out drives with multiple arms and platters to reduce arm contention. After all, if that customer file is covered by three arms rather than just one, it will take one-third the time to move the read/write head into position.

NICS

When it comes to network interface cards (NICs), SBT favors pragmatic concerns over those of sheer performance. 3Com has offered an Etherlink Plus card for several years now, but at a considerably higher cost than its bread and butter half-slot card. The newer card's prime benefit, offloading network drivers from DOS memory, was aimed at 3Com sites using that vendor's LAN operating system. After migrating from 3+, its appeal faded. Novell in turn provides a high-performance 32-bit Ethernet card for the EISA bus (NE3200) that caught deGrazia's attention; he uses it in his most recent acquisition, a 486 EISA server. But in every other node he uses 3Com's mainstay, the 8-bit 3C503. "Its a battle-hardened card," he says. "Anything that can go wrong with it already has. And it's economical because it lacks a coprocessor or on-card memory." Although deGrazia says he plans to take a closer look at the newly announced 3Com 16-bit card, he is satisfied for now with what he's got.

Multiport Repeaters

deGrazia is also reluctant to make any change from Ethernet. "Wiring is something you shouldn't change too often," observes deGrazia. LAN personnel get to know the old cable runs and their associated problems." That sort of continuity helps SBT's MIS identify cable problems in the making.

Indeed, deGrazia feels the biggest deficiency of the Ethernet bus topology is its vulnerability to cable breaks. While IBM's Token Ring is equipped to map out problem cable segments and the associated workstation, the Ethernet bus design depends on a continuous flow of data packets down a straight wire., The problem is much like the old

"Christmas tree effect": when one decorative light in a set hooked up in series burned out, the entire string went dark.

Two 3Com multiport repeaters provide some protection against this by insulating other segments of the LAN from problems on one run of cable (see Figure 1). Should a coaxial connection fail in, say, the marketing department, only the users along that cable segment will lose access. With eighteen segments, SBT has minimized the risk of failure due to a cabling disruption. Segmentation simplifies the process of locating cable disruptions, too, because it greatly reduces the section of cable that needs to be examined.

Bridges

Another potential problem with a network as large and active as SBT's is total traffic on the wire. Heavy activity in one department can affect LAN performance for the whole company unless the net is segmented using bridges.

SBT has created two "sub-LANs" to isolate the heavy network traffic produced by Quality Assurance and Technical Support. Traffic from such intense technical activity and test data traffic are sequestered on these network subsystems, reducing overall activity on the larger network. To afford occasional access to SBT corporate data, these two "sub-LANs" are bridged into the corporate system. Two 3Com bridge boxes (one for each subnetwork) are connected to their own cable nodes. Data packets originating from Marketing that are destined for Technical Support go through a repeater, enter the Technical Support bridge, and are fed through that department's file server to the intended destination. The Quality Assurance system is configured in the same way (see Figure 1). Aside from an occasional query by Technical Support staffers about SBT customers, activity tends to stay on their side of the bridge. Conversely, business activity such as end-of-month transactions doesn't bog down technical personnel.

Troubleshooting

No matter how well configured to prevent trouble and provide adequate performance a network is, there is a potential for trouble. But when trouble comes at SBT, they have to do far less guesswork to tack it down. Over the past year Harris became sold on network analysis tools, based on consultants' visits with Network General's Sniffer. This Compaq 386-based device monitors Ethernet activity, displaying packet sizes and transmission characteristics. After leasing the Sniffer for a few months, SBT bought a more economical alternative from a local VAR, Enterprise Network Systems: The Network General Watchdog (about $2,000 compared to the company's flagship Sniffer at $22,000). The Watchdog is a hybrid of hardware (a circuit board) and software (an analysis program). It monitors network activity and provides graphic charts and reports. SBT is pleased that Watchdog provides 90 percent of the capability of the Sniffer for ten percent of the cost (or, if you include the price of an inexpensive 386 clone to house Watchdog, less than 25 percent of the cost).

Watchdog occupies a prominent position—the monitor on the Watchdog equipped node faces passersby in the main corridor. "It's so valuable a tool to us that we keep the machine on 24 hours a day," says deGrazia. "It will be the early warning system when and if cable problems emerge." Its location makes it possible for MIS personnel to quickly examine it while passing by throughout the day. "We've even had users approach us and remark, 'Boy, traffic is usually high today'," deGrazia says.

Documenting LAN traffic is a weekly activity with a slew of textual Watchdog reports including lists of the most active file users, "talkers" (workstations sending the most packets) and "listeners" (those receiving the most packets). If a performance problem emerges later, historical trend lines should be very helpful in pinpointing the source.

A crew of overzealous carpenters gave Watchdog a workout last year. In the wake of the October 1989 earthquake, San Francisco Bay Area businesses have been urged to secure all fixtures. Workmen, diligently tightening fastened bookshelves, ran a screw through network cabling in Product Development. Watchdog called out the problem. A segment of thin Ethernet cable was run and they were back in business ten minutes later.

Another tool for diagnosing LAN and even application problems is LAN Assist from Fresh Technologies. LAN Assist is a 9K TSR that allows a network administrator to take remote or shared control of the keyboard on any given workstation. For training purposes, both administrators and users can alternate between keyboard control and a pop-up "chat window" within which they can communicate. deGrazia liked the approach so much he extended it to offsite access, equipping key MIS people with PC Anywhere and fast modems so they can diagnose and solve network and user problems remotely. It is possible to run Watchdog even from home.

Tuning Applications

Being a software developer, SBT realizes that tuning applications software also yields results. Lessons learned on the headquarters LAN rebound to the benefit of VARs and end users of the company's accounting software.

Whenever possible, deGrazia tries to use local workstation drives; they're typically underutilized. With FoxPro/LAN, each workstation is configured to write scratch files to a local disk directory assigned with the DOS environment variable SET TMPFILES=. This simple measure improves performance at the workstation while reducing traffic on the cable, yielding better performance for everyone.

The interaction of hardware and software also has an impact. Over the past year, the proportion of 386 workstations rose from ten percent to 50 percent. This gave SBT a chance to exploit 386 memory management. All 386 machines are equipped with Quarterdeck's QEMM-386, permitting each 386 node to load Novell device drivers into high memory. After placing IPX and NET3 there, users win an extra 40K of DOS memory. Harris reports dramatic performance gains from FoxPro and dBASE IV as a result of that extra DOS RAM. Because better workstation performance means locks are held for shorter periods on the servers, overall throughput is improved as well.

deGrazia makes extensive use of NetWare 386 RAM caching features to speed index accesses on the servers. As a result, most index searches are performed in memory, preventing more expensive disk I/O operations. Harris and deGrazia eagerly await the release of FoxPro 2.0; the compacted index files will make it possible to further reduce hard disk thrashing.

Backup and Data Redundancy

Backups are performed in various ways. Besides the typical incremental tape backup regimen for all files on the network, SBT has added time stamps to transaction files, and exploits these to perform hourly backups of key data. They don't resort to exotic technology to get the job done. Files are indexed on the time stamp data, then FoxPro's COPY command directs new records to a mirrored drive across the network.

While data redundancy is achieved today by keeping copies on duplicate drives, SBT is about to roll in a Retrochron Vortex. The Vortex is a mirroring subsystem that includes a multiplexing disk controller and a 1.2 Gigabyte drive. What appeals to Harris is the Vortex's built-in transaction tracking and rollback features. This subsystem continuously logs deltas of all disk writes, making it feasible to restore images of files as they existed at any moment in time.

With all this technology, why continue to do tape backup? Although mirroring will protect against a disk crash, Harris realizes it is not a panacea. In the case of a catastrophe, it's entirely possible that all devices in a chassis (or even a building) will be equally devastated. That's why backup tapes are stored off-site.

Future Directions

To goose up performance, deGrazia is on the verge of installing intelligent routers to replace the multiport repeaters now in place. These devices improve throughput by managing where data packets go. Multiport repeaters simply take incoming data packets, amplify the signal, and rebroadcast the transmission across all network segments. The intelligent router knows which network addresses are found on the various cable runs it manages and targets its transmissions. Each cable carries only those packets destined for an address on that segment. Data traffic is significantly reduced. In effect, routers bring to the Ethernet bridge topology benefits offered by star and token ring topologies: isolation of cable segments and reduced collisions. The result should be higher data transmission rates.

There's still more SBT wants to do. Individual cable segments within the Technical Support and Quality Assurance LANs can't be differentiated by Watchdog under the current configuration: The bridge devices obscure details on those cable runs. The remedy is simple: Install Watchdog units within both of the bridged networks.

Pushing and Sharing the Envelope

"I'll bet that one-tenth of one percent of all Fox network installations do as much as we do," says deGrazia.

SBT president Davies concurs. "We're pushing the envelope, but it's real interesting when you realize that the envelope holds the entire company." He knows SBT has more than its share of server crashes from being so far ahead of the curve. But the bright side, a very bright side, is that as they push the envelope, they find solutions that can later be exploited by the more mainstream LAN sites of SBT's customers.

Davies thinks all big LAN sites should cooperate to solve such problems. He'd like to exchange experiences with other corporate LAN installations. The collected data could help to bullet-proof the LANs of tomorrow. And deGrazia sees hope on this front. Novell has dedicated two people to do nothing but catalog data on big corporate sites: "Their information gathering is meticulous and they're willing to share." Meanwhile, Harris says SBT is among Fox's top 100 multiuser sites. He hopes to start a "Century Club" on CompuServe for Xbase sites exceeding 100 nodes.

If SBT's efforts to pool networking intelligence succeed, the benefits are likely to reach thousands of nodes near you.

Vendor Addresses

- 3Com Corp., 5400 Bayfront Plaza, Santa Clara, CA 95052; 408-764-5000.
- Fresh Technology Group, 55 W. Hoover Ave., Suite 9, Mesa, AZ 85210; 602-827-9977.
- Network General, 4200 Bohannon Dr., Menlo Park, CA 94025; 415-688-2700.
- SBT Corp., One Harbor Dr., Sausalito, CA 94965; 415-331-9900. ■

Evaluating Network Service Providers

In this report:

Datapro Summary

Ten years ago, most mission-critical computer applications were developed by in-house MIS personnel, possibly working in conjunction with a large data processing consultancy. In today's environment of heterogeneous network computing and quickly changing technology, it is more important than ever to find quality providers of network computing services. Consultants and other service providers offer a variety of network computing services. They specify hardware and software purchases, develop custom applications, and train staff. Additionally, some consultants provide network administration services. For example, if a company wants to install a PC-based network to provide an application for sales, it could hire a systems integrator to specify the equipment, customize off-the-shelf applications (or develop new applications), specify other consultants as experts in certain areas, and train staff and users. Although the mix of services and areas of focus that external service providers offer varies by company, there are specific issues to consider when hiring help for any network computing-related services.

Why Use Service Providers?

Organizations face the reality of needing outside help for service, training, and support for several reasons.

Massive corporate downsizing and layoffs that affect information systems budgets. Corporations are unable to maintain expert information services staffs, and formerly large information services budgets have been scaled back. Using outside service providers is one way to keep projects moving along without incurring all the costs of maintaining full-time staff.

Difficulty in locating, hiring, and maintaining a skilled staff. Today's networks are heterogeneous collections of hardware, software, and connectivity products. Finding and keeping staff with experience with the latest technology, and maintaining their skill levels, is difficult and expensive.

Executive dissatisfaction with corporate IS departments' failure to provide competitive advantages through the use of computing technology. Until the 1980s, some corporate information services divisions were given nearly unlimited amounts of money. Massive computing projects were undertaken in the hopes of gaining competitive advantage through the use of technology. Competitive advantages have been attained through technology, however, by only a small percentage of organizations.

Quick pace of change makes it difficult to keep up with new technology. New computing products are announced daily. Sorting through marketing claims and promises is a full-time job in itself. Businesses that hire external service providers can focus on their line of business, instead of computer and information systems.

—By Katherine Wollerman and Richard Scruggs
Business Systems Group, Inc. (BSG)

Fear, uncertainty, and unwillingness to take responsibility. Implementing mission-critical information systems can be extremely risky. Using consultants allows corporate decision makers to transfer some of the risk. The service provider assumes many of the risks associated with information services, including risks involved in acquiring new technologies.

Temporary needs for hard-to-find, highly skilled personnel. Building and installing corporate information systems requires more personnel than maintaining the systems. Because it supplies technicians on an as-needed basis, external network support increases a company's flexibility and efficiency. A company does not have to increase and decrease its MIS staff as projects do the same. For instance, if 200 nodes need to be installed on an existing network, an outside provider, such as a network integrator, can be used. The network integrator will assemble enough technicians and project managers to do the installation within a specified time frame. Network installation requires many people, whereas network maintenance does not. By hiring an outside firm to install a network, the extra hands needed for the short-term installation need not be hired permanently. Handing over short term work to another party frees the companies' employees to concentrate on ongoing projects.

MIS departments and end users often have conflicting views on how things should be, making the service provider's third view valuable. Politics is another reason for using outside resources. Some service providers are better at dealing with computer hardware and software vendors, and have a keen sense of what is vaporware and what are proven solutions. This is especially important when multiple vendors' technologies are integrated, or when a LAN must be cured of symptoms caused by product conflicts or multiple potential problem sources.

Determining Costs

Using external service providers can provide cost savings at many levels, but determining the exact cost differences between internal and external services is difficult. Although some costs for specific network service and support jobs can be measured, many are elusive. Various costs are unaccounted for, such as the considerable amount of maintenance that can be performed by the user (estimated to be from 15% to 30% of a user's time), or support services contracted for by managers but never documented specifically in budgets and reports.

In addition, the cost of LAN downtime can be very expensive. Customers spend an estimated annual average of $3.48 million in lost productivity, $606,000 in lost revenue, and $60,000 in direct expenses. These costs should be factored into cost/benefit analyses; they should also be factored in as penalties in contracts.

These costs include:

- Training and acclimation of new hires—fitting them in with the company and business
- Providing specialized training
- Providing benefits packages—health and insurance plans, FICA, social security, worker's compensation
- The on-going obligation and effort by most companies to find and keep a place for employees

Types of Service Providers

The major network services providers include manufacturers, distributors and resellers, and independent consulting organizations.

Manufacturers

Manufacturers of network computing products, like Novell, IBM, Digital Equipment Corp., Apple, and Sun, offer consulting services to their customers. Novell's Advanced Technical Services (ATS) group is an example of such an organization. The primary benefit of using vendors' consulting groups is that these groups are intimately familiar with their products. Of course, many people believe that the vendor consultants are too narrowly focused on their own products. Additionally, they are relatively expensive, few people know about them or how to best use them, and they concentrate primarily on very large accounts or operate on a very limited basis as technical adjuncts.

Distributors and Value Added Resellers (VARs)

The use of distributors and resellers as service providers is common. Resellers operate under the premise that they can add third-party consulting value to a client during an installation. Resellers are often more accessible than a manufacturer and offer a turnkey solution for installation, training, maintenance, and support. A drawback to this approach is that resellers, like manufacturers, are biased toward the software and hardware they sell. To reduce this bias, some resellers can be hired for their consulting services only under an arrangement where they are barred from bidding for the actual equipment.

Independent Consultants and Integrators

Independent service providers are organizations that do not sell hardware or software; they sell only the professional services of their staff. Independents can be regionally or nationally based, and can range in size from a single consultant to tens of thousands of employees. National consultants have multiple offices across the country and range in revenues from $20 million to $5 billion. These firms are positioned to support large, complex networks that span the world. These firms have extensive resources and support services to justify the large investment. Regional networking consultants have from one to a few offices, usually within one geographical region, and typically have revenues up to $20 million.

An advantage of using independents is that there is usually less bias toward any given product, although some integrators specialize only in some areas. An additional benefit, especially prevalent in small to medium-sized providers, is a high level of expertise and experience in "making everything work together smoothly." Most independents claim to go beyond mere technical functionality to address computer-based solutions to business problems and requirements.

Organizations should approach their relationship with an independent service provider cautiously, however. First, the organization must be assured that the service provider is experienced enough to take on the project. References should be obtained from the service provider's previous clients. Potential clients must be satisfied that the service provider possesses the expertise that it claims. Potential service providers should have a successful track record, as well as experience with the systems and products that the organization is using.

Next, a contract should be drawn up specifically detailing each party's responsibilities for the project. The service provider should be given measurable goals for each stage of the project, and penalties for nonperformance should be included. An unsuccessful system implementation can severely damage an organization's business.

What Service Providers Can Do

Services offered by consultants and integrators vary. Some will do everything from start to finish; others specialize in one phase of the networking process, though they may oversee the entire process. Some commonly offered services include:

- network planning, design, installation, and testing
- applications development, installation, and customization
- user and administrator training
- network security and integrity auditing
- network administration, troubleshooting, diagnostics, and repair
- product specification and procurement
- overall integration

Case Study

The following case study illustrates the relationship between a real company that hired an independent systems integration consulting firm to develop systems that would help them gain a competitive advantage.

What is now a large independent gas marketing company began in 1984 as a start-up firm in a quickly evolving industry. This company soon found itself in a hot marketplace. They had hired a dozen employees, purchased six IBM PC XTs, and written a few Lotus 1-2-3 applications.

To gain a competitive edge, the company contacted a regional systems integration consulting firm. The integrator saw that the gas marketer was growing quickly and needed a system solution that could keep pace with its growth. An overall informational flow of the business was developed, along with an initial software and hardware strategy supporting the flow.

The integration consulting firm suggested the company use a completely integrated system that would support the basic functions of the business: gas marketing and exchange, gas balancing, and general accounting. These applications, in addition to the entire network configuration, (including hardware and software inventory, cabling, and communications to remote locations), were designed and overseen by the integrator, although the actual physical network setup was performed by a LAN installation firm. The consultant also evaluated several development options and simultaneously managed the development of the company's own internal staff.

The gas company's new computer system soon became a major factor to its growth. Because of the strong and flexible system, the company has significantly expanded the original system to accommodate unexpectedly rapid growth without having to hire excessive systems-support personnel.

Other Issues

By choosing external integrators and consultants, many risks are avoided. Still, some decision makers, especially those that have been burned in the past, are leery of relying on external help. Critics point out the value of having an in-house team: career paths in a company's organization are created for individuals, service is implicitly built into the arrangement, and workers tend to be more willing to do extra work. Depending on the situation, it may be difficult or very expensive to get work done that is not specifically defined in a contract with a service provider. There is also strategic value in having your own talent and expertise in many areas.

Managers are often afraid they will lose control of a project if they rely completely on outsiders. There are four key issues to consider when looking at the problem of control: operational control, strategic control, control over the future, and disaster control. The key to success in the preceding case study was that the company did not rely 100% on the integrator to solve business problems. Companies that turn to external service providers with expectations of instantaneous resolutions to information services problems, through activities like long-term outsourcing, are unlikely to achieve their goals.

In the natural gas marketer example, the expertise of the integrator was certainly necessary for success, but even more important was the integrator's willingness to spend time considering what the real needs of the company were at the time, and what they would be in the future. A strong, mutually successful client/consultant relationship was formed.

Choosing a Service Provider

When entering a relationship with a service provider, it is important to know up front what you want to gain from it. Carefully examine existing procedures and systems, and ask fundamental questions about your business problems.

Does the service provider have strong vendor ties? Does the service provider specialize in PC-based solutions, or does it provide a mix of mainframe, minicomputer, and PC-based solutions? Is the service provider tied formally or informally to any particular product or products? Remember, these ties can be either a benefit or a drawback.

Has the service provider been successful in past ventures? Does it have experience in situations that match yours, e.g., your industry, your size, your business problems.

What services does the service provider offer? Are you looking for a single source for all of your information services needs, or are you seeking specific assistance with one project. Do you need training services or ongoing support of your corporate systems?

Is the service provider certified in specific technologies or products? Some manufacturers of computer hardware and software and professional societies offer certification programs for third-party suppliers of networking services.

Can the service provider support your network in every city in which you need support? Does its size match your needs?

Are network services the service provider's primary business? If so, the service provider probably has a diverse range of talent and expertise.

How will you divide support tasks between in-house and contracted expertise? How large a role do you want the service provider to play in your business?

How are services priced? Are services offered on a pay-as-you-go or fixed rate basis?

What response time do you need on problems and how does the service provider define response time? This is particularly important when contracting for administrative and repair services.

What spare parts does the service provider keep in stock, at your site, and how fast can it get parts? Ready availability of spare parts is often critical.

This report was developed exclusively for Datapro by Katherine Wollerman and Richard Scruggs of Business Systems Group, Inc. (BSG). Katherine Wollerman is a consultant with BSG SI Consulting's applications development group. Richard Scruggs is BSG's director of business development. BSG, based in Houston, TX, is a national systems integration company specializing in business solutions based on client/server, network computing technology. BSG can be reached at (713) 965-9000.

Summary

Consultants, systems integrators, and other network services providers can lend key assistance in achieving business goals. To achieve the best results, however, it is vital to understand exactly what services the provider offers. It is also essential to find an integrator or consultant that understands your business problems and will work with you to find the best solutions in an environment of mutual trust. Understanding how the consultant provides services to your firm also helps you gain a better understanding of the demands and possibilities that network computing offers your organization. ■

End-User Computing in the LAN Environment

In this report:

Datapro Summary

End-user computing, already common before the era of local area networks (LANs), is growing in importance as LAN-connected workstations become the norm for user access to computing. A LAN gives users the powerful, friendly tools of the PC along with easy access to data maintained by corporate systems running on other platforms. Users can build more systems, and systems of greater importance, especially ones that extend and enhance corporate systems. Information Services (IS) must facilitate this development and deal with problems of system quality, data integrity, and security that are common to end-user systems. To handle these issues in the LAN environment, IS must staff the end-user computing support function with a wider range of skills than is typically available today.

Introduction

End-user computing is not a new phenomenon—it predates both the personal computer and LAN eras. As PCs connected to LANs have become the standard in more and more organizations, however, new opportunities for end-user computing present themselves, accompanied by new management and control challenges. Today's users, equipped with powerful workstations hooked via LANs to a wide range of programs and data, are increasingly willing and able to take responsibility themselves for meeting their computing needs.

As a result, a LAN today is likely to host not only a larger number of end-user systems than appeared on mainframes or standalone PCs, but also systems that are more critical to the organization's function. LANs, therefore, intensify a challenge that has always accompanied end-user systems—that of facilitating effective use of the technique while controlling potential problems. For Information Services (IS) management this means ensuring that needed tools, data, and staff support are available, while at the same time helping users avoid the common pitfalls encountered in developing and running computer systems. Many organizations will have to augment the resources they devote to end-user computing support in order to fill an expanded range of responsibilities.

—By Lee Gremillion
Price Waterhouse

Defining End-User Computing

The term end-user computing covers a range of activities, and is best understood by first considering what it is not. At one extreme, end-user computing is more than parametric use of canned systems. A parametric user is one who invokes functions provided by the developers of the system and supplies parameter values. Everyone is a parametric user to some extent—we fall into this category every time we use a bank's ATM. We invoke functions such as "withdraw" and "transfer," and supply parameters such as a password and dollar amount. Most people working with ordinary transaction-processing systems in organizations are parametric users.

At the other end of the spectrum is professional system development; that is, design and construction of systems by professional analysts and programmers to be used by others. Traditionally, IS professionals have built systems for parametric use by their clients—staff in the production, marketing, finance, and administrative wings of the organization. End-user computing implies that the client, not an IS professional, develops and maintains the functionality. Sometimes the term user development is used as a synonym for end-user computing.

The end-user computing landscape between these two boundary markers is broad and varied. Just across the line from parametric use, we find users tailoring canned software to fit special circumstances or needs. At the other extreme, we find users designing and writing programs from scratch using general-purpose languages. Local area networks do not change the range of this landscape, but they make it richer. Being on a LAN increases both the opportunity for end-user computing and its value by making a wider range of software tools and organizational data available.

A Historical Perspective

The impetus for end-user computing has traditionally been the backlog in applications development. Backlogs—queues of application development requests awaiting action—are as old as IS departments. Application system development has always been a matter of expanding demand chasing scarce resources, and promises to remain so for the foreseeable future.

In the mainframe world, where end-user computing first appeared, the most common way for users to develop their own systems was with fourth-generation languages (4GLs). A 4GL could be a self-contained language (such as FOCUS or EASYTRIEVE), or it could be part of a data management system (such as IDMS or TOTAL), in which case it might be called a "query language." It would typically provide for data storage, maintenance, extraction, analysis, and reporting. Its specification language would be nonprocedural, or would be at least at a higher logical level than the typical third-generation language (e.g., Cobol).

Most of the user-developed systems in the mainframe world have been periodic reporting and ad hoc retrieval functions to supplement existing corporate systems. The complexity of the mainframe environment limited what users could do for themselves, independent of IS professionals. It was relatively easy for IS to monitor and guide end-user computing, since IS "owned" both the tools and the data on which they operated.

The personal computer explosion of the early 1980s shifted the focus of end-user computing. PCs were inexpensive and, above all, simple to use. The premier PC software package, the spreadsheet, gave users a general-purpose tool based on a familiar, comfortable concept. PC users quickly learned that spreadsheet packages could be used to model far more than accounting statements. As subsequent generations of spreadsheet programs added ever more sophisticated features throughout the 1980s, users responded by applying them to an ever widening range of applications.

Along with spreadsheet software came database systems for PCs—sometimes integrated with spreadsheets, sometimes standalone. These scaled-down versions of mainframe data management systems gave users the tool they needed to design, build, and maintain their own databases on their PCs. Word processors and graphics packages were also used in innovative ways to provide functions that otherwise would have had to come from IS. The standard, friendly system software of the PC even made it more attractive to a minority of end users to write programs in such languages as Basic, Pascal, and C.

From the user point of view, a great benefit of the PC was that it enabled the user to operate independently of the organization's IS function. One could get the PC, spreadsheet, database, and other software, and build the application without ever having to wait in the IS backlog. In fact, the backlog was often not the issue—it was just easier and more natural to build a system directly than to try to have it done by IS. This independence, however, had a negative side—the organization could all too easily end up with many disjoint islands of technology, running incompatible, often low-quality systems, storing redundant, often inconsistent data. Standalone PCs freed users from the constraints of the eternal scarcity of IS application development resources, but at the cost of isolation from corporate systems, data, and standards.

As LANs have become increasingly common, end-user computing has developed a new impetus, since the LAN gives users the best of both worlds. They can still employ inexpensive, easy-to-use PC software tools to build their applications, without requiring a professional intermediary. Data, on the other hand, can be stored and maintained once at the most logical level (corporate, departmental, local) and shared across the LAN to whatever applications need it. There are still potential problems, such as quality, consistency, and security, that must be addressed. If the problems are avoided, however, LAN-based end-user computing can play a vital role in an organization's use of information technology.

User-Developed Systems

The range of user-developed applications is as wide as the range of functions within an organization. It is possible to categorize end-user systems into three broad groupings: ancillary, departmental, or individual systems. This categorization is meaningful because the nature of opportunities and problems, as well as the support role IS should play, varies from category to category.

Ancillary Systems

Perhaps the most common example of end-user computing, ancillary systems are applications that do not stand on their own, but rather supplement corporate systems. They may be used to format raw data into useful reports, combine data from several sources for reporting purposes, or perform calculations that are easier to implement and maintain locally than to get programmed into a corporate system. Users sometimes build ancillary systems because the backlog in application development keeps IS from enhancing the core system quickly enough. In other cases, they add function (e.g., presentation graphics) that could not be easily built into a core mainframe system.

Anyone attempting to replace a corporate system is likely to find a thicket of ancillary end-user systems that must be accommodated by the replacement. For example, one large insurance company recently found no fewer than 34 separate spreadsheet-based end-user systems attached to its mainframe investment accounting system. A LAN that includes platforms running corporate systems will only spur the end user's drive to build ancillary systems, since it makes data access easier. The quality of such systems is an important issue, since they are extensions of vital corporate systems, often themselves providing essential functionality. IS should play a direct and active role with ancillary systems to help users deal with quality control and data access issues.

Departmental Systems

Departmental systems are primarily standalone, but serve a limited clientele within a particular "department" (which can be any organizational unit). Both spreadsheets and database systems are commonly used for this purpose—to store records not needed elsewhere, or in a more usable form than those stored in corporate databases. Budget planning and tracking within a sub-unit is an application commonly automated by means of a user-built departmental system.

The two evils that can plague departmental end-user systems are redundancy of function and redundancy of data. Redundancy of function means reinventing the wheel, when one group of users spends resources building a system to address a problem for which a satisfactory solution already exists. Data redundancy is a problem because redundant data (e.g., the same data stored on multiple PCs and/or the corporate database) inevitably become inconsistent. Both problems should be much easier to solve if users participate in a LAN than if their computers are strictly standalone. The IS role in departmental systems involves coordination of functional and data resources, as well as help with quality control.

Individual Systems

Individual systems serve the needs of a single person in the organization. For example, a manager might use a database program to keep records of expenses for a project that are more current than those kept by the corporate accounting system. These systems typically serve very specific, even idiosyncratic needs, and are often quite short-lived. The LAN affects this type of end-user computing primarily by making more tools, faster hardware, and a wider range of data available to individuals. The key to making individual system building most effective is education, so that users know the tools and data available and how to use them. The primary IS role is to provide training and consultation, a role with which many IS staffs are both comfortable and adept.

Glossary

Ancillary System. An end-user system that supplements or adds functionality to a professionally developed corporate system.

Database Management System. System software that provides facilities for the organization, storage, maintenance, and retrieval of data.

Departmental System. A standalone end-user system that serves the needs of the members of some organizational unit, such as a department.

End-User Computing. The development and maintenance of application software systems by persons who are not IS professionals for use by themselves or the organizational groups to which they belong.

Fourth-Generation Language (4GL). A general-purpose programming language that is non-procedural or at a higher logical level than procedural languages such as Cobol.

Individual System. An end-user system developed by a single individual for use by that individual alone.

Information Center. An organizational unit the purpose of which is to support end-user computing.

Parametric User. A person whose use of an application software system consists of invoking predefined functions and supplying parameter values expected by those functions.

Scripting Language. A language for specifying commands to control the operations of one or more application software systems.

Management Issues

End-user computing and user development of applications systems have always presented an array of thorny issues for management. Operating in the LAN world simply raises the stakes in some cases, because the end-user systems on a LAN are likely to be more complex and important than those found on standalone PCs. In other cases, the nature of the issue changes as we move from the mainframe or standalone PC environment to that of the LAN. In many cases, a "LAN manager" function, separate from individual end users, is needed to deal with the issue.

Development Standards and Quality

Users generally have not been trained in the disciplines of software engineering, and are likely to make mistakes as they develop systems. Problems typically include:

- writing programs, procedures, or specifications that do not function correctly, primarily because no disciplined approach to analysis, specification, and testing is used
- using the wrong data on which to base calculations or reports
- omitting controls and audit trails so that errors cannot be detected and corrected

These mistakes are costly whether the system runs on a mainframe, standalone PC, or LAN. In general, the cost of these mistakes to the organization rises with both the number and importance of end-user systems in place. The software and data facilities of the LAN foster increases in both the frequency and importance of end-user computing, thus making this issue loom larger in the overall IS picture.

Backup and Data Integrity Issues

Systems operated by IS normally include mechanisms to assure the integrity and recoverability of data. Databases are copied to backup media regularly, and access to maintenance functions are limited. Any competently managed IS operations group can recover with minimal data loss from disasters ranging from unintended deletions to massive hardware failures.

The end-user track record in this area has not been as robust, although it is improving as basic computer literacy improves. In the standalone PC environment, users are responsible for their own protection, but when a problem occurrs, the damage is limited to a single PC. Shared file server use in the LAN world complicates this picture. Multiple users are likely to be affected when a hardware problem such as a disk failure occurs. In addition, files stored on the file server are likely to be larger, more complex, and more important than those stored locally. A mistake or failure that would be an inconvenience on a standalone PC could be a disaster if it happened on a LAN file server. Ensuring that file servers are adequately backed up (i.e., protecting the users in spite of themselves) is one of the primary functions of the LAN manager.

Security

This is a particularly thorny issue in the LAN world. Typically, security was not a major issue with mainframe-based end-user computing, since a database administrator (or similar function) controlled access to data for everyone. In the standalone world, physical access control—that is, locking up machines and diskettes on which data was stored—usually sufficed for security. On a LAN file server, the physical hardware that stores the data is shared among many users. In many LANs, however, there is no database administration role to control access to data. Someone, almost certainly the IS function, must educate users about security issues, and provide (and enforce) security mechanisms. This also must be a LAN manager responsibility.

Hardware/Software Evaluation and Selection

For standalone PCs, central guidance of hardware and software acquisition served primarily to control cost. Central purchasing might obtain better prices for both hardware and software. The range of both hardware and software had to be kept within the bounds of what was reasonable to maintain and support. Within these constraints, however, users selected products independently according to what best suited their particular needs.

A LAN adds a new level of constraint to the process, since all parts—hardware and software—must work together. Once a LAN architecture is selected, someone must determine whether hardware and software under consideration is compatible with that architecture. Until the day of universal standardization arrives, this will require a central evaluation agent. Since many of the questions involved in making such a determination are technical in nature, the IS function is the obvious candidate for this role.

Controlling Redundancy

Redundant data inevitably becomes inconsistent over time. Redundancy is unavoidable when users develop systems on standalone PCs, since each PC must have its own copy of the data it is using. Consider, for example, a spreadsheet-based investment analysis system that operates on securities holdings extracted and downloaded from the investment accounting system on the mainframe. The PC data matches that on the mainframe at the point the extract is done, but, as soon as new trades are executed, the extracted data is no longer current. This problem can be managed, since it is easily visible. Much more difficult to manage is the case where multiple systems are gathering and storing the same data independently.

If redundant data can be identified, connection to a LAN offers a means of eliminating most of it. For one thing, if files can be shared on a file server, there is no longer any reason for every PC to have its own copy. Mainframe data that must be extracted and downloaded can be done so automatically, regularly, and probably more frequently. The ultimate solution is to have data stored only once, at the most appropriate level, available to any process on the network that needs it. While this ideal (distributed databases) has yet to be achieved in most computing environments, shared data on a LAN can provide a much better solution today than the widespread redundancy found in standalone PC systems.

Documentation and Transferability

When users develop their own systems, they often fail to take the time to write the documentation necessary for the system to be used and maintained by others. Writing documentation is tedious and time-consuming, and the person who just built the system does not need the documentation to run it anyway. The problem usually surfaces when someone new must use the system, and the original developer is not present to explain its operation.

This is not an issue for individual systems, since their lives typically coincide with the time they are useful to the individual who developed them. Ancillary and departmental systems, on the other hand, often serve constituencies beyond the person or persons who build them. Multiuser ancillary and departmental systems running on LANs need usable documentation no less than do the corporate systems developed by IS and running on a mainframe. In the long run, good documentation protects the investment made in developing a system by making it easier to use and change. IS should educate users to this need, suggest documentation standards, and help with documentation tools and techniques.

The Role of IS

The widespread use of LANs changes some management issues in end-user computing, and generally increases the importance of them all. LANs also increase the value of end-user computing in the organization, and give IS new motivation and opportunity to address the issues. To do so, however, IS must invest in an active program of end-user computing support. This requires personnel resources, education and training, and management commitment. For many organizations, this will mean changing and upgrading the way they support end users.

The Information Center: What Is There and What Is Missing

Organizations most commonly support end-user computing via the Information Center (IC), a concept first formalized by IBM Canada in 1974. The notion was that users could satisfy some part of their computing needs more directly and rapidly if the right support was available. The IC would address this need by providing education, technical support, and help with identifying and using tools and finding data.

Originally in 1974, the IC was to be a physical place where users could get access to mainframe terminals and printers, as well as obtain consulting help with their use. Today, in organizations in which users are connected to LANs via PCs, the IC more commonly provides consulting and support, but not physical resources. Recent studies show that IC services fall into six categories, listed here in decreasing order of the frequency with which they are offered.

Hardware support includes assistance with evaluation, selection, installation, maintenance, and problem solving.

Functional support is assistance in identifying and specifying the application needs, and assisting the user in designing and building or acquiring the software needed to address it.

User Development Tools

The range of discrete software packages that could be used on a LAN to build end-user systems is enormous, numbering at least in the hundreds. These tools can be grouped into several major categories, each of which provides a different, if sometimes overlapping set of functions.

Spreadsheets
Spreadsheets, the software that made the PC popular, are used for an incredibly wide range of applications. Spreadsheet models have been written for such diverse functions as production scheduling, investment analysis, pension calculations, and remote data entry. The base concept is a matrix of rows and columns, but modern packages offer features far beyond the spreadsheet itself. Report writers, macro or procedural languages, statistical functions, and graphic output extend the spreadsheet's capability to serve as a general-purpose tool.

One feature particularly useful in end-user computing systems is the ability to separate models from data, so that a given model can be run against different sets of input data. Many ancillary systems are spreadsheet models that are run on a regular basis as new data is pulled from corporate databases or external feeds. For example, one bank built a spreadsheet model to calculate fees due on international securities trades, and runs it daily to process data extracted from an external trading support system.

Database Management Systems
These are currently second only to spreadsheets in the frequency with which they are the basis for user developed systems. Database management systems provide a range of facilities for the definition, population, maintenance and retrieval of data. Most modern systems feature nonprocedural methods for retrieving and reporting data, as well as a procedural language for writing "programs" to access the database. Both end users and IS professionals have used database packages to build complete application systems, some quite complex.

LAN-oriented data management systems add a feature not available on their standalone PC predecessors: they can provide for access to data stored on other machines. In its most primitive form, this means easy file import and export facilities. At the sophisticated end of the scale are facilities for real-time data interchange between databases situated on different platforms. These facilities enable LAN-based users to build cooperative processing applications by adding local functionality to existing systems running on mid-range or mainframe platforms. For example, one mutual fund company captures information about prospects on PC screens built using a PC database system with a graphical user-interface screen-formatting facility. The data is then shipped to the mainframe to be loaded into the shareholder database.

Scripting Languages
A script is a sequence of commands to control a particular computing environment; a scripting language produces scripts. The net effect is a facility with which users can extensively tailor one or more software packages to meet custom needs. A simple example of a scripting language is the set of commands that can be placed in the PROGMAN.INI file to be executed when Windows is initiated. At the complex and comprehensive end of the scale is Rexx, a system procedure language for IBM SAA strategic computing systems, including OS/2. Some scripting languages (such as the macro facilities within Word and WordPerfect) work within a particular application; others (such as Rexx) are designed to control multiple applications.

Scripts serve to make standard application packages more useful by customizing them to meet particular user needs. For example, a script might be used to automatically run several applications to pull together data from multiple sources for reports. The script would contain commands to log into several corporate databases, download selected information from each, and invoke the PC database or spreadsheet software to produce a particular series of reports from the extracted data. It might be set up to fire off automatically each day at a specified time, or to initiate when a certain event occurred.

High and Very High-Level Languages
The term high-level language was originally coined to distinguish such languages as Cobol and Fortran from the hardware- oriented, "low-level" languages that preceded them. The high-level language used by most end users has been Basic, although it has been eclipsed somewhat by derivatives of C and Pascal. High-level languages are procedural; that is, a programmer must describe the program's function step-by-step. They are also general purpose, not tied to a particular database or spreadsheet package.

Very high-level languages resemble the 4GLs used on mainframes and are generally less procedural than high-level languages. Using a very high-level language, a user typically specifies the results that are desired, rather than the steps required to obtain them. A prime example of a nonprocedural, very high-level language is SQL, which is rapidly becoming the universal query language for relational databases. Some very high-level languages stand alone, while others are bundled into database systems. Very high-level languages (particularly SQL) often serve as the means by which a PC program or user interacts (over the LAN) with a file server or mainframe database.

Software support includes assistance with package evaluation and maintenance, software library maintenance, and application program maintenance.

Training and education includes training on system and application software, and procedures, such as security.

Data support includes maintenance of a data dictionary, assistance in locating and transferring data, and providing archiving, backup, and recovery mechanisms.

Miscellaneous services such as coordination to prevent redundant systems, network support, and a hot-line.

For the most part, these functions are appropriate whether the end users are building budgeting systems with spreadsheets on PCs or writing reports with a 4GL on the mainframe from data in a corporate database. In two areas, however, it appears that IC services in many organizations are lagging the move to LAN use. First, fewer than 25% of ICs today maintain a data dictionary to help users find what data is available where in the organization. This was a less important function when users either used mainframe tools against corporate databases, or built standalone systems. With the connectivity of the LAN, however, the power of end-user computing can be enhanced greatly if users know what data is available, where it is, and how to get it. The IC role must expand in this area, since it is truly a cross-organizational function, beyond the realm of any one user department.

The second service with a current level of offering inappropriate for the LAN environment is security. Fewer than 15% of ICs today provide security training for their clients. Again, this area has been of less importance in the mainframe world (where security was controlled centrally), or with standalone PCs (where physical access control provided security). In the LAN environment, the physical resource (file server) is shared, and users are typically expected to be responsible for their own security. LAN operating systems provide the mechanisms needed to implement security, but it takes training and management to strike the happy balance between protection and sharing that is needed to make the LAN and its file servers most effective.

The information center can continue to play an important role in end-user computing even as the environment evolves. The goal is still to help the users solve business problems in a timely and effective manner. How the IC is organized and staffed, and what services it provides may have to change as the computing environment changes.

Staffing for the New Roles

The traditional IC support for end-user computing has been strongly oriented toward helping individual users with specific hardware and software questions. The archetypical example of an IC function has been to help a user staff member design a spreadsheet program to do the group's budget. Seldom did anyone raise the question of whether this was the best way to address the application, or whether this was the right application to address.

The need for this type of support has not disappeared, but a LAN-based user community demands a broader, more sophisticated level of support. As we have seen, the widespread use of LANs increases both the opportunity and the challenge of end-user computing. In general, users are developing applications that are more complex and more important than in the past. These systems, particularly the ancillary and departmental systems, become an integral part of running the business for many users. End-user systems have become big business for many organizations—they consume significant hardware and organizational resources to perform important functions. To help with these forms of end-user computing, the IC staff must combine a strong foundation in software engineering with an understanding of the user's business. The emphasis must be on how to use the best combination of products and data to effectively address business needs.

To play this widened role requires a shift in what has been the focus for many IC staff. Traditionally, a good IC staff member was one who knew the ins and outs of PC products and was good with people. Product knowledge and people skills are still important, but they must be supplemented with broader knowledge in several areas.

Understanding Corporate Goals and Objectives

The IC staff should help users determine whether building a particular application is the best use of resources, or whether there are better opportunities. To do this, IC staff members must understand the organization's critical success factors and its most important information needs. If, for example, customer service makes or breaks the company, the IC staff should promote end-user development of customer service-oriented systems, and downplay or even discourage efforts in other, less important areas.

Knowledge of Core Corporate Systems and Data Resources

To be effective, end-user computing must fit into the overall corporate information processing architecture. Having end users build a customer records system, for example, that duplicates functions and data already provided by the corporate order processing system is wasteful and counterproductive. IS professionals are in the best position to help users understand what is (and is not) provided by the corporate systems. This help is best delivered if it is bundled with the other services provided by the IC.

Awareness of Range of Technology in Organization

Too often today, PC people and mainframe people form two completely disjoint sets. End-user systems in the LAN environment are increasingly likely to call for cooperation among platforms—some PC, some midrange, some mainframe. To support development of these systems, the IC staff must know what is available and what is best handled on each of the platforms the organization utilizes.

Skill in Systems Engineering Practices

As LAN-based end users take on increasingly complex and vital applications, the role of the IC in helping with development quality becomes more important. IC staff members must translate the systems engineering standards and techniques devised for large-scale systems projects into practices that end users can adopt to improve the quality of their systems.

An individual proficient in all these areas simultaneously would not only be the ideal IC staffer, but would be in great demand for a wide variety of IS tasks. In practice, not every individual on an information center staff must embody all these skills in order to be effective. The important point is that IS management recognize how end-user computing, and, therefore, its support needs, are changing, and ensure that the IC has the pool of skills needed to provide that support. ■

This report was prepared exclusively for Datapro by Dr. Lee L. Gremillion, Senior Consulting Manager in the Boston office of Price Waterhouse. His professional experience has focused in the area of systems planning and development, especially using CASE tools and advanced methodologies. Before joining Price Waterhouse, Dr. Gremillion served as head of a software development firm specializing in rapid application development using CASE. He has consulted and lectured widely in the U.S. and abroad on application development techniques and methodologies, and has published articles on these and other information systems management issues in such journals as *Harvard Business Review* and *Communications of the ACM*. He received his doctorate from the Harvard Business School.

The Strategic Information Management Challenge

[redacted]

In this report:

This report will help you to:

- View strategic information management as a service, not as technology management.
- Utilize strategic information management to add value to your business enterprise.

Introduction

One theme developed in this report is that it is basically wrong to think of IS as technology management. We are not fundamentally a technology-management activity, we are fundamentally a service. These ideas give us some difficulty, because the management schemes we use inside the IS organization tend to follow the models of a manufacturing business and not of a service business. Yet many issues can be addressed by not attacking technology management but by adopting some management values drawn from service businesses. Indeed, our problem is a relationship problem in just the way that services have relationship problems with their customers and clientele.

This Datapro report is a reprint of "The Strategic Information Management Challenge" by Robert J. Benson, pp. 26-45, from *Journal of Critical Issues in Information Processing Management and Technology*, Volume 7, Number 4, 1990.

Because we have historically dealt with things as technology matters, we tend to think that the commonality of technology gives us insights that we can transfer from enterprise to enterprise. We commonly share discussions in which companies stand up and say, "It's like this," or "No, it's like that." Because we are managing a common technology, we can talk meaningfully across our organizational lines and experiences and learn something; this is always the unstated assumption. This has led us to misunderstand, that as a service business, we are dealing with different kinds of clientele. Our businesses are different.

For example, we hear much about financial institutions, where the system is the business, and the insights derived from that may or may not work in an enterprise where the system is not the business, but merely is an aid to managing the business. One fundamental message in this report is that the commonalities that we should be deriving from each other come, not from the commonality of the technology but from the commonality of the business units we are serving. This leads us to a second problem.

We tend to think of these things as enterprise matters, but that is a fallacy because each of our enterprises is also complicated, engaged in many different businesses. So if we accept the premise that we are a service business, that we have different kinds of clientele, and that the key common factor is the characteristic of the business, then we must be capable of analyzing the enterprise we serve in terms of its different kinds of businesses. I will give you an example from the university. People do not like me to talk about the university because it sounds non-real. It is difficult to imagine where that impression came from, because we are a fairly robust-sized organization, so we tend to evidence some of the same characteristics of a real business.

Thus we have different customers, so it is not useful to talk about strategic planning for "the university." It is useful to talk about strategic planning for each of the major business components or lines of business, such as undergraduate education, as different from research. Indeed, those are two different businesses, and everything is different about them. No wonder we fail when we try to talk about strategic planning for the enterprise if we do not take into account that we have very different business activities that we are trying to serve. More importantly, the idea of success is different in those businesses.

An unbroken chain of logic is becoming apparent. Though it is important, the key issue is not the technology management. The key issue is service to business clientele. The characteristics that differentiate one enterprise from another, and within the enterprise, are based on business considerations, not on technology considerations. This forces us to look at how to describe the business. We call this diagnosis. Diagnosing the business is needed to permit us to organize our IS services to have a maximum effect or impact on that business.

My second point is that we are a cost, rather than an asset, when we are not perceived as a part of whatever it takes to move the performance of the business. What moves the performance of the business is that all other aspects of the business are related to the lines of business, such as marketing, production, and customer service. Yet, we have typically not thought of ourselves as particularizing our activity along those same differential lines of business.

For example, I run a computer organization that serves the undergraduate line of business, the research line of business, the health care line of business, and the overhead of the institution. I treat everybody as though they were exactly the same. They have the same system-development methodologies for those very different lines of business. They have the same charge-out scheme. But if I were in a real service business, the idea of imposing a common charge-out scheme across extremely different business users or in different stages of their development would be ridiculous. Yet, that is what we do, we have common architectures. The whole orientation I have historically taken, and apparently it is common to other IS groups, is that things are the same for everyone.

The PC is not the driver of decentralization, instead it is the decentralized responsibility for business performance. The PC may be an engine which permits that. However, no one would care if, in fact, the asset they had available to support improvements were a part of the central controlled and planned resources, uniquely identified and specified for each business unit or line of business. It all fits together.

One key word we have developed for the Center for the Study of Data Processing is amateurization, particularly of computing, with some of the decentralization that is getting the users involved. That is a very powerful idea. It is the sense that the skill sets being applied are becoming less over time, largely because the pool of people is much broader. The problem is more fundamental than that. When we talk about managing the business that we are in, which we believe is a service business, we are amateurs. It is not a question of taking business knowledge from our business, but of knowing how to manage the service as a service business, and not as a manufacturing business. Some thematic ideas that will be discussed include diagnosing the business characteristics of our enterprise, managing ourselves as an IS service, and the concept of strategic information management. What is important in this idea is that it is moving the performance of the businesses we are serving.

In our undergraduate line of business at the university, the key performance measure of the line of business is the quality of the student. In our kind of institution, we are not competing on price, but on quality. Implicitly, the performance measures are on the quality of different classes. The better quality we have, the more attractive we are to students who would come to such an institution.

Everything is measured on the performance of the undergraduate line of business, on average GPAs, on what percentage of the students are in the top ten percent of the graduating high school class, and that sort of thing. In that context, the investments in IS, thought of as being assets to those running the undergraduate line of business, are not those assets wanted for delivering financial information or student records. Rather, they are related to those things that will move the performance: the recruiting systems, the instructional support systems, and so forth.

We have invested many years in the world's best student-record system. Our student-record system is so good that our students could change their grades from their home. But when that is measured against business performance of that undergraduate line of business, those systems are totally irrelevant. That is a problem, but it emphasizes this idea of what we are trying to get done.

Add Value to the Enterprise

We at Washington University, together with the IBM Scientific Center in Los Angeles, have had a joint study in Enterprise-wide Information Management since 1984. Much of the work presented in this report has resulted from that joint study.

The Strategic Information Management Challenge is the challenge of adding value to the enterprise.

We are talking about the things driving the mentality of the business domain and, by extension, should be driving our mentality. We are not talking about such things as architectures, delivery systems, and structured methodologies. These, in the wrong hands, are probably the disasters that will cause us not to be very successful and certainly not thought of as assets.

When we talk about adding value to the enterprise, we are actually referring to two different topics. One is how to obtain the best business value. That means we know what it is. What is the basis for business performance? The second is, how to apply the best approach to add value. This involves the view that the measure of what we do is not based on technology terms but on business value-added terms. In turn, this is based on moving the business performance numbers for whatever the lines of business we are engaged in. We should not be doing anything that is not up to that measure.

Couching it in the context of thinking of our work as a service, the key question is, "How do we add value?" In much of our work, we have found it helpful to divide the universe into the business domain and the technology domain. The technology domain is the part that we manage, and the business domain is a set of customers. The problem is that our set of customers have historically been well-buffered from the real performance of the business. We have a history of working through the overhead of the enterprise. Actually, we are facing how to add value, not viewed from what the value systems are (Figure 1, lower section), but viewed from the perspective of the upper section of Figure 1, the customer domain. Our business performance measures are those that relate to how we, in fact, perform with regard to that customer domain.

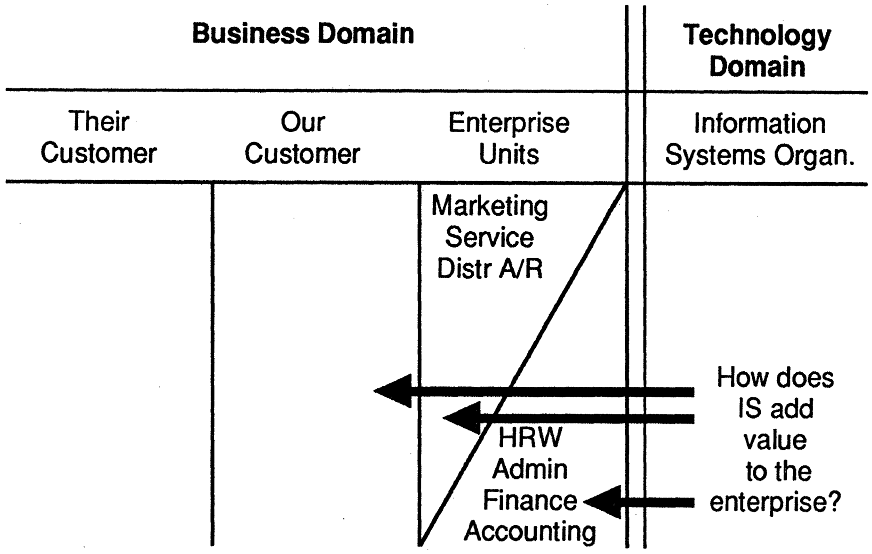
We will discuss this in six parts:

1. We will cover the IS challenge of managing the technology domain.
2. Where business value comes from.
3. IS as an addition to value.
4. IS as a service business.
5. The action plan. None of these ideas make any difference if we cannot do something about it. That will be the only measure of this report. Can we actually apply any idea that might come out?
6. Finally, you will be brought up to date as to what we are doing to further develop and implement these ideas in real companies.

An MIS Challenge

The challenge we have is to back away from our origins. If we had discussed strategic planning for Information Technology even five years ago, we would have tended to look at three clusters of activities: data, hardware/software systems, and delivery systems. We would talk in terms of applying

Figure 1.
The Domains of Information Management



an architecture to these. We would talk in terms of constructing these things in a framework, or a blueprint, of activities that we should be doing. These are important, but they are part of the same level of importance as building the manufacturing plant. It is simply not the same thing as saying, this is the set of technology planning that will drive our business, which is serving our enterprise, for maximum business effect. This is certainly an appropriate collection of things to do, but it is second order. The first order is to respond to the classic management questions: How can we compete better? How can we improve our internal productivity? How can we motivate the entire enterprise to think about these things? We will probably fail if we expect business management to care at all about this technology planning. There are some exceptions, of course, which generally arise in enterprises where the system is the business. That is a key diagnostic difference. If the system is the business, then these technology issues become the business-planning questions.

My basic belief is that you must be able to diagnose the character of the business domain to understand these things. When I say that technology is second order, I mean that it is for those enterprises where the system is the business. What is important are the drivers. From our standpoint, this has meant two things: One is that we add drivers, not from enterprise strategic planning, but from line-of-business strategic planning (Figure 2). Enterprise strategic planning tends to focus on financial outcomes, portfolio management, and things of that kind. Line-of-business strategic planning tends to focus on moving the business measures, whatever they may happen to be. Some are financial, some are market, some are product, and some are quality. The exception is in single-line-of-business companies. It is quite frustrating that most of the successful examples of strategic planning described at the enterprise level are actually single-line-of-business companies masquerading as an enterprise.

Steering committees are a disaster at the enterprise level where there are multiple lines of business. They are a requirement at the line-of-business level because that is where the business performance is and where all of the other assets are being determined, planned, motivated, and mobilized. The thrust of our approach should now be apparent; diagnose the enterprise, discover the kind of businesses you are in, and move from that as a driver to doing these things.

We see four clusters of things that represent strategic information management in the search of things we are starting to understand: business performance with the link-to-technology planning, business as the driver with line-of-business being the unit of the driver, and for each of those the strategic plan for the line of business (Figure 3). The effects and how they are carried into performance measures represented in the organizations that have the strategic plan will be defined. This point is crucial, but we must understand, as we diagnose the business, the performance measures of the productive elements that carry out the line of

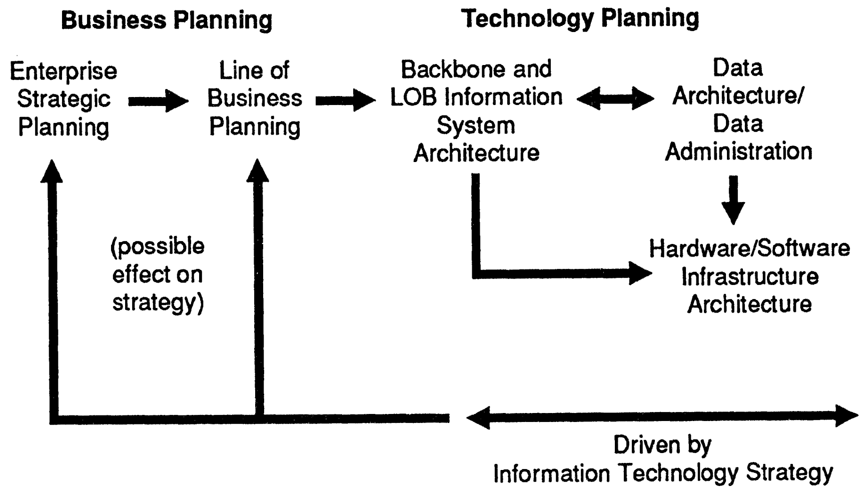


Figure 2.
Planning Results

business. Our information systems and our technology planning can affect strategic planning. In a way, we see a circular relationship in Figure 3.

The IS challenge is to put the technology management into the second order of things and to put our understanding, our diagnosis, and our recognition of the business issues as the first order of things. It is our problem, because we are in the service business. If we do this, we will see a shift in emphasis. We will see a shift from application orientation to data orientation: a shift away from backbone, that is overhead, to architecture, meaning the set of things that support the organization and plan for the lines of business in which we are engaged. There will be a shift from technology to business, from enterprise to line-of-business, from operations to strategy. If you think about it, this is what we have taught, trained, and motivated our current staff to be concerned about.

This is what we should be teaching, training, and motivating our services staff to think about, and the list is upside down in terms of order of importance. Strategy at the line-of-business level is driving the architectures producing the data for the line of business. That is the nature of the IS challenge.

Business Value

We found it helpful to separate the enterprise domain into the two parts: the business domain and the technology domain. IS management and strategic planning for IS is really managing the interface and the relationships between these domains, like a service business. A little more complicated than that is the first generic enterprise model which we offer (Figure 4). Models are important in talking about this because we really have no agreed-upon principles in our business. Although 15% of the Gross National Product is tied up in information processing in some way, we do not have any agreed-upon vocabulary. We cannot talk to each other, except in vendor-enabled quasi-standards. We need models.

The key part to this enterprise model is on the left. Most enterprises actually work by dividing themselves into what they think about as corporate-level things, and line-of-business-level things, organized in terms of business units. There are many of those. Corporate views itself as making money available to the business units, such as operating money in budgets. It might be capital money.

Whether we need it or not, they also make available policy and guidance. Some are different

Figure 3.
Business Domain/Technology Domain

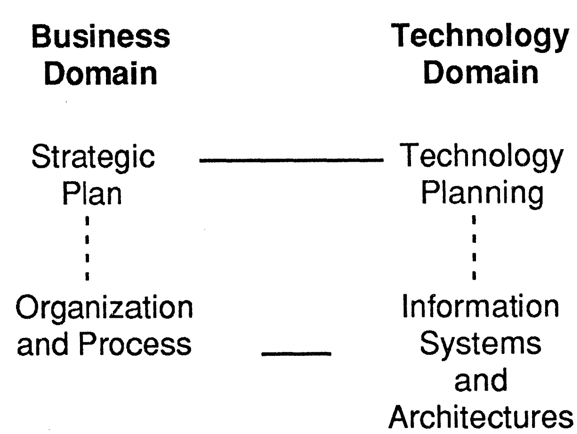
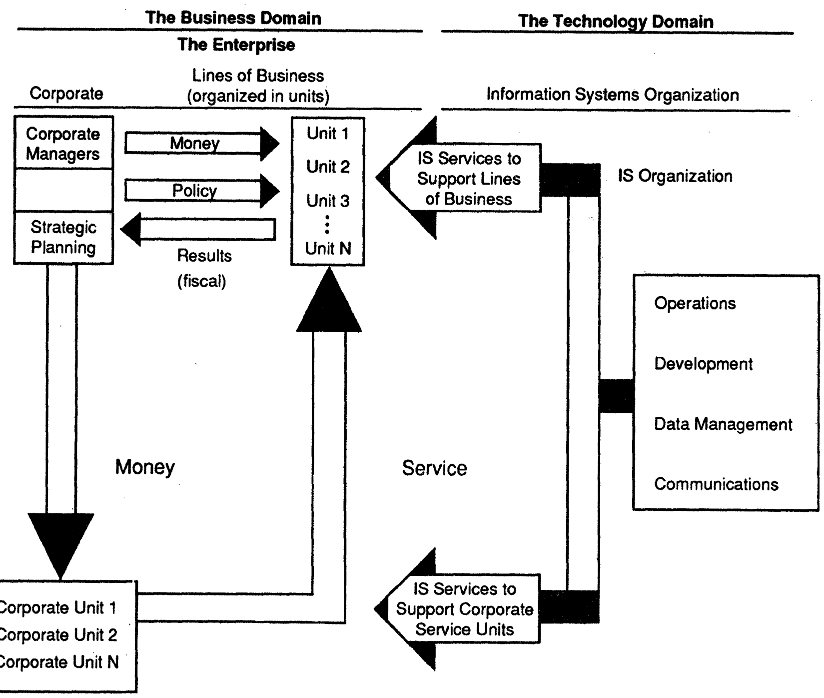


Figure 4.
The Enterprise Model I



from others but the framework must be understood. What they get back from us is results. Corporate-level financial planning, or strategic planning, is really looking at that process. What policies and frameworks do we need? What investment and budgetary constraints do we have? What do we expect back? Of course, we also have some corporate units like accountants and lawyers, and all they get is money. Possibly we get some service. They forget that sometimes, but it is appropriate.

From the IS standpoint, we must do everything. We provide services to the lines-of-business and to the corporate overhead units. We are basically in four service businesses: operations, development, data management, and communications. Each has a service component to it. Our problem is to look across this interface to the enterprise we are servicing.

One key idea when we look carefully at the enterprise model (Figure 5), is that when we are talking about the line of business and business performance, it is the relationship between the business units and its customers, products, and services. The real question is, What are we doing in IS that moves the business performance with regard to its customer needs?

Michael Porter has been extremely helpful in this regard. You are probably familiar with his work on corporate advantage, corporate strategy, and the value chain. These are powerful ideas that help us diagnose the business unit. Porter discusses the value chain as one enterprise feeding another by selling products and services, and so forth. His work has permitted us to understand the basic character of business value, of what we must move to be an asset.

With Michael Porter's permission, the value chain will be changed (Figure 6). This version is composed of our enterprise and our line of business, buying goods from our supplier, and doing things for or to our customers. Our customers have a choice: to buy from us or from them depending on their business value. There are three categories of things we can affect to add value:

1. Affect customer choice.
2. Improve our performance once that choice has been made (that may be quality, or that may be cost).
3. Improve our financial performance.

The fundamental test of the IS service organization is that, if the things we are doing for our enterprise do not measure up to one of those three added values, we are in the wrong business.

The key issue is, do we know the businesses that we are servicing? Do we understand what the basis for performance is? More importantly, does our service business understand it? Do those who

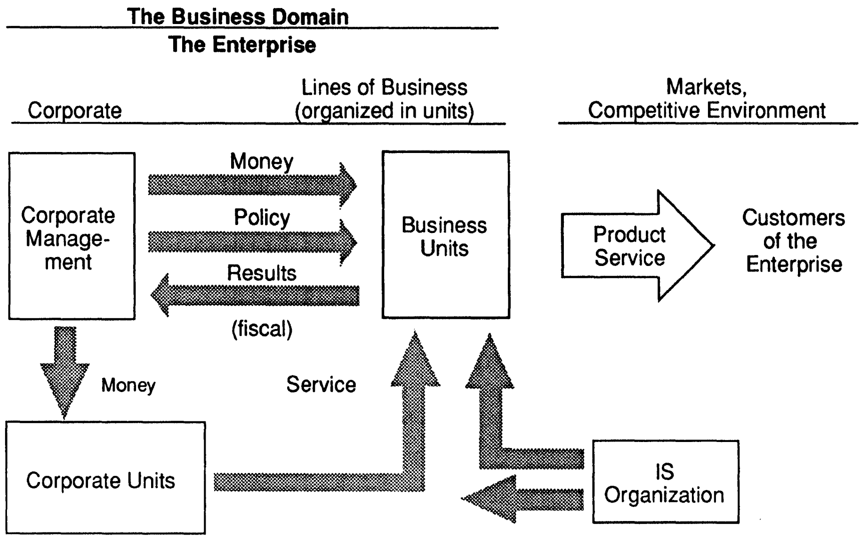


Figure 5.
The Enterprise Model II

are running our operations service business understand? If we view ourselves as being in the manufacturing business, or as running a technology, then all we care about is our own internal performance. Manufacturing firms also have this problem. But, in fact, we have a service business, it is to see not just our performance, but our effect on our customer's performance.

It gets to be very difficult to do this. There are many crucial differences among businesses and lines of business (Figure 7). It is our assertion that lines of business value fall into three fundamental clusters: product, service, and government. In clusters around product-type lines of business, the performance is based on customer choice, the way in which we deliver product to customer, and the quality in financial performance, three basic measures. By extension, we can take these three ideas and ask a line of business, how do you measure those things? And by extension, we can say that IS is adding value if we move those measures. We can measure customer choice, and such things as market share, delivery, time, quality, and cost performance.

The business value for a service-oriented line of business is still customer choice. It is not delivery, but response to customer requirement. How do we measure response and performance? We use the word government to mean any nonprofit, noncustomer-choice situation. The business value is not customer choice, but it is now access to service. It is not delivery or response, it is capacity to respond, and it is still business performance. Incidentally, it is interesting that in our own internal IS service businesses, the operations business and the communications business are much like this. The systems development business is very much like service. The data business is very much like product. When you analyze the IS business along these lines, each of those four fundamental businesses has very different characteristics in terms of the measurements we would like to use as to our own internal business performance. The key is to understand, to diagnose, and to take action based on our understanding. And the understanding is very simple. What business are we in, and how do we measure our performance? By moving those measures, we make a contribution, and hence are of value and are an asset to the enterprise.

Planning Using the Seven Questions

One approach we have developed is something called the Seven Questions (Figure 8). We apply these seven questions to the enterprise and to the

Figure 6.
The Future of Information Management I

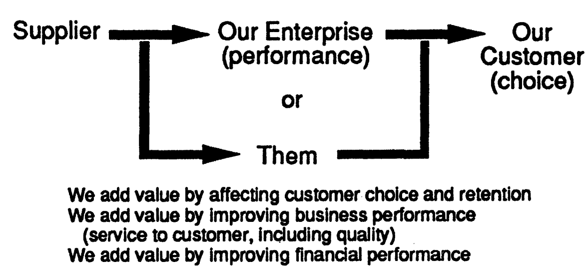


Figure 7.
Crucial Differences

Product:	Choice, Delivery, Perform
Service:	Choice, Response, Perform
Government:	Access, Capacity, Perform

IS organization, separately. We should think together about how we would apply these seven questions to our own enterprise. Can we meet the test of being able to answer them? They are very simple questions. Question one is, What business are we in? That is actually a question of what lines of business are we in? What different customer sets do we have? What different product types do we have? And are we fundamentally a product or a service or a utility type of business? All the measures are different.

Question two is, What problems do we have? What problems do we have in improving those performance measures? The definition of a problem is: What stands between us and the business in moving those numbers?

The third and crucial question is, What strategies do we have for doing something about it? We now have characterized the business part of the planning somewhat differently from how we usually think of traditional strategic planning. This says explicitly, that success in our planning will come from moving the measures. By extension then, the strategic planning comes from removing or enabling solutions to the hurdles that stand between us and that. These are very powerful ideas. Instead of being abstract, they become concrete.

In our university undergraduate line of business, what business are we in? We are selling quality education to parents. What are our performance measures? Market share, quality of student, break-even financial performance. A slightly longer term is alumni donations. What are the problems in moving numbers that represent those basic performances? The pool of eighteen-year-olds is going away. There is much competition. When you think in those terms, you begin to realize that we messed up our strategic planning eighteen years ago. But the real key is, what hurdles do we have getting the word out about recruitment to the right kind of students? You can begin to see, when one conveys these three basic questions about what business problems we have and what strategies we are following, why I can argue that a wonderful student-record system and that a terrific financial system might not be relevant in a strategic context, in an asset context. The idea is to do the seven-question kind of planning with the business managers. This is why the steering committee is important at the line-of-business level. The agenda for IS investment must come from recognition of these three basic questions.

Question four is, now that we know what business we are in and what problems we have, What will we do about it with Information Technology? This becomes the basis of information for IS as an asset in the business. We must improve the functioning, achieve success, and that means enabling these strategies.

Question five is, How can Information Technology be used to change the strategies? The two

Figure 8.
Planning Using the Seven Questions

1. What business are we in?
   - product
   - customer
   - strategy
   - choice
   - performance
2. What problems do we have?
   - competitive
   - customer
   - performance
   - opportunity
   - technology
3. What strategies are we following (in response)?
   What opportunities do we have?
   - management
   - initiatives
   - compensation
   - incentive
4. How can Information Technology be applied to improve the functioning of the business, and achieve success in its strategies and critical success factors?
5. How can Information Technology be applied to change the business strategy, or change the basic business plan (new product, customer, market)?
6. How can we organize the Business to most effectively achieve its goals, business strategy, and plans?

   How can we organize the Information Technology activity to best accomplish the business goals?
7. How do we develop an action plan to get the results from questions 1 through 6?

Figure 9.
Summary

Purpose of IS is to add value to the enterprise.

IS does so by contributing to enterprise success.

Enterprise success is in two performance measures:

—Financial performance
—Business (or organization) performance

What is necessary is a planning and decision process that identifies basis for performance, and produces improvements in performance.

questions here are to enable current strategies, and to contribute new strategies.

Question six is, really, How can you recognize the Information Technology activity needed to best accomplish those business goals? Given our origins as technology managers, who treat everyone alike, using common methodologies and everything else, and viewing ourselves as a manufacturing business, rather than a service business, you can imagine that we have some problems here that we must overcome. It is probable that we are dysfunctionally organized in the context of measuring up to this challenge of moving business performance.

The seventh question is, How do we develop an action plan to get the answers to questions one through six? In the technology domain, we have the same essential problem. Apply the seven questions to us. What business are we in? What problems do we have? What stands in the way of business performance improvements? First is defining the basis of our performance as a service business in each of our four major components. Second is developing our own strategies for doing something about our own performance measures, and, simultaneously, deciding how we will attack the problem of dealing with our multiple lines of business that are our "service set," our service customers. Thus, the questions of chargeback schemes, methodologies, organizational forms, information centers—all the traditional things—are measured against what strategy we are using to deal with each line of business. A fundamental message is that no one is exactly the same. It is madness to construct the same processes for everyone, if we want to be successful, because the measure of success is in their performance improvements, not ours.

MIS as an Addition to Value

How do we do all this? First, an intermediate summary is given in Figure 9. The purpose of IS is to add value to the enterprise. That is a very fundamental statement. Secondly, IS does so by contributing to enterprise success. Third, enterprise success lies in two kinds of performance measures, financial performance, and business or organizational performance. Fourth, what is necessary is a planning and decision process that identifies the basis for performance, and introduces improvements in performance.

In the business domain we need the process of applying those seven basic questions or their equivalent (Figure 10). I say "equivalent" because I have given them generically, but they obviously will be different industry-by-industry, business-by-business. Every business is different. We must be able to figure that out, and diagnose what we are doing. Why the seven questions? They are the process by which we will define business performance in terms of measurements. Define the answer to question three, which is business strategy. What are we doing about our hurdles? The action plan is for doing something about business performance and then using information economics as the basis

Figure 10.
Business Domain Action Planning

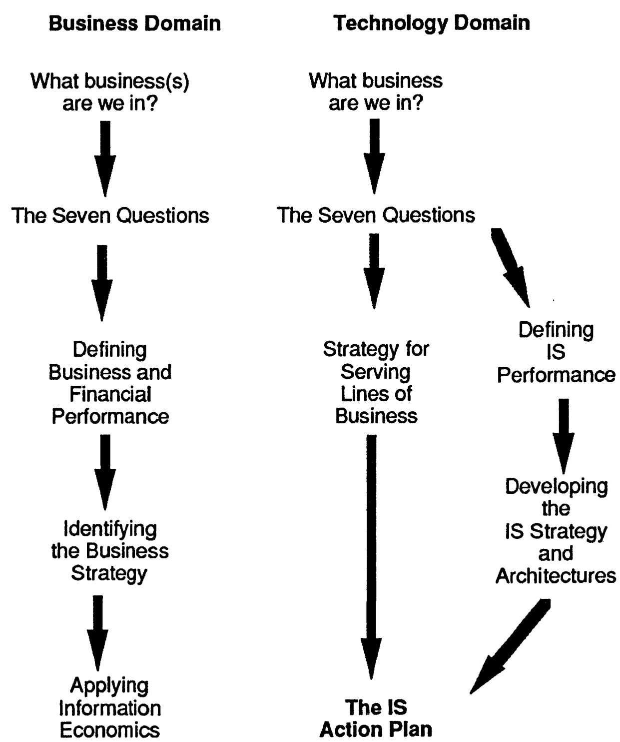


Figure 11.
The Future of Information Management II

Manufacturing Model	Good = decreased unit cost Back-office values Standard Economy of scale Specialization
Service Model	Good = increased service Front-office values Custom Effectiveness of size Generalization

for decision-making across projects. Do they measure up to adding value to the enterprise?

MIS as a Service Business

We now have a single cohesive vision of how to go about doing strategic planning to add value. There are only a few minor problems with this: politics, reality, history, money, reshaping our own internal processes, and technology. Just a little collection of details. But the message is, "Unless we start, we are never going to get there." The first place to start is to think about what business we are in. The history of IS has been that we are a manufacturing business. We manufacture systems, communication networks, and standard PC workstations. Assemble may be a better word. If you look at how one manages a manufacturing business, many value-systems have come along (Figure 11). A good one is decreased unit cost. A good way is to separate front office from back office. There is specialization of function. That is part of the critical mass of doing the manufacturing. There is economy of scale and there are standards. That is the way we have basically organized ourselves over time.

If you look at it as a service model, the goodness is in increased value of the service. Quality service is that which the customer will pay more for. One of the problems we have with IS managers is that our services do cost more than the perceived cost of the PC or the minicomputer. A "goodness" is also front-office values. There is customer contact, customer service, solving the customer's problem with a personalized solution. This is not standard, it is custom. Effectiveness of size brings to bear your resources in a service organization through a single point of contact. Generalized capacity, not specialized capacity is important. It is generalized to respond to any problem that arises.

You see that there are some fundamental conflicts. To add business value, our perspective as a manufacturing model is a catastrophe, unless we go toward the service model. It is not easy, but we are part of that problem. We are the people who go up there and mumble about our architectures for data, systems, and communications and build the perspective that we think of this way. We are part of that problem, and we are talking about reversing a 25-year mind-set. Cultural change is needed. Some of the things we think are important enough to tell our users are these words out of a manufacturing book: Zero defects, reusable code, reducing of unit cost, critical mass, economy of scale, standardized things, and standardization.

We are doing much work trying to understand the service business as a generic thing because we are all amateurs. I am an amateur at understanding how to manage services. It is surprising when you look at the business literature. There is so little about managing a service business. There is not much of a systematic nature. This is surprising since 70% of the work force in the United States is in some way engaged in the service business, yet we have this kind of gap between ourselves and understanding the businesses we are serving, and so little understanding about how to manage ourselves.

At Washington University we are doing an important sequence of what we call "strategic workshops in managing the service business." Unlike some academic institutions, I am not persuaded that we know everything. In fact, there is some possibility that we know little, but we surely have the right questions. It is a place to start. For example, when we did the first of these strategic workshops in St. Louis, we tried to define service (Figure 12). We are getting closer. A good is an object, a thing; a service is a deed, a performance, an effort. The essence is the performance rendered by

Figure 12.
Definition of Service

Definition of Service

We will adopt Leonard Berry's definition that a "good is an object, a thing; a service is a deed, a performance, an effort. Although the performance of most services is supported by tangibles, the essence of what is bought is performance rendered by one party for another." (1984) Two contemporary criteria are most often used to distinguish services from manufacturing organizations. The first is that the output of services is intangible, the second is the closeness of the consumer to the product (Fuchs 1968).

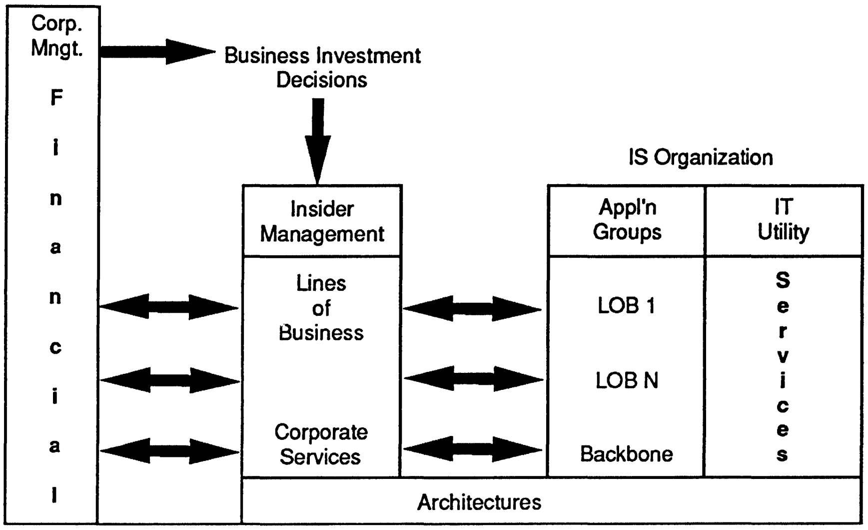


Figure 13.
Corporate Planning Model

one party for another. It is intangible, the closeness of the consumer to the product. That is our business, is it not?

We are working very hard to further this basic notion that we fundamentally are a service business. There is some possibility that systems development has some of the attributes of manufacturing, but in the age of purchased software, that is not crucial. In some of our consulting work, we are working hard in the service business to define quality of service—it is successful response to client demand. When we apply that to our own business, you can see how that has some power. Quick, effective, sensitive, knowledgeable, creative, and individualized.

Why do we consider all this? We need to focus on the service characteristics, and understand quality, and performance, and how to do it as a service business. You can take this metaphor forward and look at how we currently manage ourselves. How do we define quality of service? If we look at it from our perspective, as the service provider, we look at it as an inward matter. Availability, average of things, response time (it takes two seconds or a hundred years for whatever), and median accuracy. We look at it as providing a measure of quality. That is a manufacturing perspective. We need to understand that quality is a consumer perspective in the service business. Our concern is problem response to the customer, peak-load capacity, because that is what the consumer feels: response to customer need, ability to handle new circumstances that arise that we have not planned for with no hassle and ease of use. You see the difference between orientations.

When we take the idea of thinking of ourselves as a service business, what we think about is efficiency and effectiveness. From the provider perspective, efficiency is the service-delivery unit cost. Effectiveness is availability of the function. That is manufacturing. From a service perspective, efficiency is the consumer unit cost. How much work does it take to use what we deliver? Effectiveness is enabling their business-performance improvement. You see again the power of adopting the service metaphor. Now, these things are important, but they are second order. First order, in viewing our business as a service customer, meaning our lines of business that we are serving. The point of planning processes that we have identified here is to articulate a way to get at these things—to do something, as compared to just waving our arms about it.

We believe there are four basic businesses in which we are engaged, and all these ideas apply: We are in the operations business, providing a service of operating applications; in the data business, providing a service to provide access to our data; in the systems-development business, providing a service in analyzing design (there may be manufacturing-performance considerations there); and in the communications business, providing a utility-type service of attachability. Service can be defined as a service product. Is there a basis for competition? Yes, we compete on the basis for our own internal business performance from the customer perspective.

As we carry out this IS metaphor, we note we have in the service business multiple collections of customers that are different; that is why they are

Figure 14.
Symptoms and Characteristics of the Forces and Factors

Force Factor	Symptoms and Characteristics
Management Concern for Business Impact	Emphasis on overhead infrastructure I/S is a cost, not an asset
Forces for Dispersion	Growth of "unplanned" systems Growth of "unplanned" networks
Expansion of Scope of IS leading to Architectures	Need for data administration EDI and other cross-unit systems
Requirement for IS Productivity	Management concern for cost Management impatience for results

Figure 15.
Forces and Factors in Information Management

Force Factor	Symptoms and Characteristics
Management Concern for Business Impact	Decision-Making (e.g., Information Economics) Line-of-Business focus New I/S professional
Forces for Dispersion	Line-of-Business focus Network is the corporate asset
Expansion of Scope of IS Leading to Architectures	Architectures Technology scan, pilots Line-of-Business focus
Requirement for IS Productivity	Development environment Line-of-Business focus I/S Service focus

different lines of business. If we are a service business, then we will respond to that, not with faceless technology organization, but rather with a team focused on each line of business with the appropriate overhead being treated as a line of business. We cannot knock the paychecks out. We could, but it would probably not be wise. We are backed up by a series of utility functions like the machine room, the communications network, itself, and things of that kind.

This, then, becomes the model for thinking about how to organize the IS business and the corporate relationship (Figure 13), separating line-of-business organizations, creating application groups, an IS group that is effective in delivering, and then becoming service-delivery groups to the individual lines of business.

The Action Plan

What will we do about it? What is our action plan? We have been to many enterprises to look at the processes of IS management, and what we now tend to believe is that there are several fundamental dislocating things happening to the IS organization in four categories: there is a demand for business impact; many forces for dispersion, some of which are technology, and some management dismay with performance or wanting the control because it affects management's performance; expansion of scope problems, and so there has been some response for architectural kinds of planning; and then the requirement for productivity. We cost too much. These are the forces affecting the typical IS organization with some resulting symptoms (Figure 14).

There is much unplanned growth. There is a call for EDI and data administration. Everyone is yelling about costs. Now, the action-plan idea says that, rather than respond to these problems on a knee-jerk basis, what we really want to do is take a strategic view of the business the IS organization is in, and our relationship to the enterprise that we serve (meaning all the lines of business), and their business performance. So the question is not, "What will we do to these symptoms, in response to these symptoms?" The question is, "What is our

Figure 16.
Action Plan

Enterprise Planning

Line-of-Business Planning

Decision-Making

strategic response?" where "strategic" is defined as our vision of our business and the performance measures that we must describe how we are doing and what we will do about the hurdles that stand between us and moving those performance numbers.

We change the agenda away from thinking of this as a technology-management problem, to think of it as an IS service-management problem. That change in perspective can allow us to tailor our organization to our particular enterprise. Stop treating technology management as a commonality across all organizations; and start thinking of ourselves as serving a particular set of clients, customers, and business units according to their needs and business performances.

Some examples we might consider in response are in Figure 15 as in the adoption of a strategy that has to do with looking at how we deal with lines of business.

You will notice that line of business tends to frequently appear here. Also note how we erect management tools, with a service focus.

Overall, the action plan we recommend you consider has three basic parts: posturing ourselves (enterprise planning with what business we are in); posturing our relationships with our lines of business, our customers, people that we serve; and installing the appropriate kind of decision-making which relates the investment in IS assets to performance of business improvement (Figure 16). That is where information economics comes in. That is an action scheme, and we believe it can be done.

Figure 17.
The Action Plan Checklist

( ) Line-of-Business Based

( ) Insider in Place

( ) Account Manager in Place

( ) Sponsor

( ) Chaos/Opportunity

( ) Business-Domain Attention

( ) Technology-Domain Problem

We believe there are conditions for success. First of all, the ideas we are trying to develop here do not require a revolution, a coup, or a dislocation. We believe these are bottom-up ideas, not top-down ideas. We can make these bottom-up ideas as soon as we lose the perspective that we must treat everyone the same, and that we are dealing with multiple lines of business.

Our view is very simple. Let us find one line of business that we can use this kind of approach with, rather than thinking that we must use it for everyone across everything. We have a checklist that we use (Figure 17). When can we do these things?

We believe very strongly in the "point-of-pain principle." People will do something when the pain of doing something is less than the pain of not doing something—business-domain attention. It has to be something people feel strongly about. Technology-domain problems that we can solve.

The greatest challenge that we have is to add value. The nature of that value is in moving the performance of the business units we serve. Think of that in ways in which IS can be an addition to value. IS as a service business as a way to enable that. Also, the idea that an action plan can be, in fact, adopted along the lines of the plan of the seven questions (Figure 8) and producing the IS action plan (Figure 16). ■

State and Federal Computer Crime Laws

In this report:

This report will help you to:

- Identify computer crime offenses.
- Understand computer crime laws.
- Recognize state and federal statutes applicable to computer crime.
- Track developments in computer criminal prosecution.

This report is designed to aid investigators and prosecutors by summarizing state and federal statutes applicable to computer crime. It covers them in the following order: state penal laws, secondary state penal laws, federal penal laws, secondary federal penal laws. In 1979 prosecutors stated that existing statutes could be found to prosecute all cases of computer crime coming to their attention. The laws were not written in anticipation of high technology crime, however, and in some cases prosecution was difficult and obtuse.

In the intervening decade, the need for laws directly applicable to computer crime became even more apparent, and most states now have computer crime statutes. Two federal statutes, the Computer Fraud and Abuse Act (PL 99-474) and the Electronic Communications Privacy Act (PL 99-508) were enacted in 1986. The rapid rate at which new laws are being adopted makes it difficult for any discussion to be completely timely. The material in this section added since the first edition is based in part on a 1985 study of prosecutorial experience with computer crime performed for the U.S. Department of Justice, Bureau of Justice Statistics,[1] and on other research.[2] A broad range of state and federal laws are discussed in addition to computer crime statutes.

This Datapro report is a reprint of "Section VI: Computer Crime Law" by Donn B. Parker, in *Computer Crime Criminal Justice Resource Manual*, U.S. Department of Justice National Institute of Justice, August 1989. The original source material remains in the public domain and may be copied by others without permission from Datapro.

State Penal Laws

Legislative Response to Computer Crime

Computer crime laws vary widely in offense named, definitions, and sanctions. This disparity stems from a number of causes, including:

- Response to local concerns,
- Desire to correct specific shortcomings of existing law,
- Apparent lack of understanding of the crimes and associated technology,
- Interest in adopting computer crime laws similar to those passed earlier in other states, and
- Need to accommodate the new law to an existing statutory scheme.

Computer crime has been broadly defined as any illegal act that requires the knowledge of computer technology for its perpetration, investigation, or prosecution. Computer crime is not a single type of crime; rather, most nonviolent crimes and even violent crimes such as homicide can be committed through or facilitated by computers.

Crimes directed at computers and information media can also include violent physical attacks as well as technical manipulations. New definitions applicable to specific actions have now been included in each of the state computer crime statutes and the federal computer crime laws.

State computer crime statutes fall into four general categories:

- *Property expansion.* Expansion of the definition of property in existing state criminal statutes to include computer systems, computer programs, data, and computer services.
- *Focused scope.* A new statute with a specific focus on a particular type of crime such as debit card fraud.
- *Broad scope.* A new statute with a broad focus to address fraud perpetrated by computer manipulation as well as damage to computer hardware, systems, programs, and data stored in computer systems.
- *Extended scope.* A new statute with a broad scope, with additional coverage for denial of use of a computer system, damage or theft to computer programs, or trespass into computer systems.

The computer crime offenses covered by these statutes are further detailed below using examples of definitions from state statutes.

Computer Fraud

In Arizona computer fraud is committed by:

> . . . accessing, altering, damaging or destroying without authorization any computer . . . with the intent to devise or execute any scheme or artifice to defraud or deceive, or control property or services by means of false or fraudulent pretenses, representations or promises.

Generally, the computer fraud provisions of other states resemble the Arizona model. Furthermore, they usually apply to accessing any aspect of the computer system.

Computer Trespass or Computer Tampering

Under the computer trespass provision found in at least 22 states, a crime is committed if a person accesses a computer without authorization. This definition covers intrusion into computers by perpetrators (including hackers) through telephone circuits. Many of these provisions are similar to the Georgia statute that includes access in its enumerated activities constituting computer tampering: "Any person who intentionally and without authorization, directly or indirectly accesses, alters, damages, destroys . . . any computer . . . shall be fined . . . or imprisoned." In addition, the Florida law proscribes trespass in two contexts, separately stated: (1) offenses against intellectual property; and (2) offenses against computer equipment or supplies.

Credit Information Tampering

Some state statutes specifically proscribe computer tampering in order to obtain unauthorized credit information from a computer or to introduce false information into a computer. The Hawaii statute, for example, provides that an individual has committed a crime if:

> . . . [h]e accesses or causes to be accessed any computer, computer system, computer network, or any of its parts with the intent to obtain unauthorized information concerning the credit information of another person or who introduces or causes to be introduced false information

into that system or network with the intent to wrongfully damage or enhance the credit rating of any person.

Trade Secret Tampering

California, Florida, Massachusetts, Nevada, and Wyoming have computer crime statutes that forbid the taking of trade secrets. The provisions are very different in nature but similar in effect. The California and Massachusetts statutes are broad, addressing the wrongful taking of a trade secret in many contexts including those stored in computers. The California law expressly proscribes stealing, taking, carrying away, or using, fraudulent appropriation, and unlawful or lawful access followed by a wrongful copying of the trade secret.

Disruption of Computer Services

Another computer offense proscribed in several statutes involves the disruption (or, in some states, degradation) of computer services. The Missouri statute provides that "a person commits the crime of tampering with computer users if he knowingly and without authorization denies or causes the denial of computer system services to an authorized user of such computer system services . . ."

Technical Definitions in State Computer Crime Laws

The technical definitions in the state laws also vary. Many states have followed the definitions proposed in early federal bills. These definitions have been heavily criticized by the technical community in Congressional hearings as ranging from being too dependent on current technology to being inaccurate or irrelevant. However, they prevailed and are found in the Computer Fraud and Abuse Act of 1986.

Vestiges of the terminology used in the early federal bills appear in several state statutes. For example, the term "access," a key term in most computer crime statutes, is most often defined as "to approach, instruct, communicate with, store data in, retrieve data from, or otherwise make use of any resources of a computer." It is so defined in about a third of the states.

The use of the word "approach" in the definition of "access," if taken literally, could mean that any unauthorized physical proximity to a computer could constitute a crime. It also may derive from preexisting definitions of access in trade secret theft statutes.

Most of the statutes define "computer," although approaches to the definition vary. The most prevalent definition is that a computer is an electronic device that performs certain functions: logic, arithmetic, or memory. In addition, the prevailing definition attempts to include data, software, and communications facilities connected or related to the electronic device in a system or network. This definition could theoretically include an entire public telephone system as part of a computer that has dial-up telephone access.

Concern that the definitions are too broad and could include anything from a digital watch to the entire telephone system resulted in various exclusions. In one state exclusions are a radio or television transmitter or receiver, television camera, video tape recorder, sound recorder, phonograph, or similar device used for reproducing information in aural or visual form without changing the nature or content of the information, unless such a device is connected to and used by a computer. California excludes automated typewriters or typesetters, portable calculators, or computers used for personal, family, or household use and not used to access other computers without any further clarification as to meaning.

These varying definitions demonstrate the confusion resulting from a complex, rapidly changing technology. For example, "automated typewriter" is not a term in general use, and its meaning is ambiguous and changing as technology advances. It could be an electric typewriter, a typewriter under computer control and without a keyboard or storage buffer, a "self-erasing" typewriter, a "dumb" (without local processing capabilities) or "smart" (with such capabilities) computer terminal, a word processor, or in the future a voice-activated data input device.

Some states use as the definition of "computer" an internally programmed device that processes data. Others limit the programmed device to a general-purpose digital device. One uses the same approach with the word "programmable" instead of "programmed," which may raise a problem if a dedicated-use device, such as an automated teller machine, is used in a suspected criminal act. Further, the defined functions of computers sometimes redundantly include storage as well as logic, arithmetic, and memory. A few states and the U.S. Congress in the federal computer crime laws, apparently contemplating technological developments beyond the electronic, expand the adjectives applicable to the device that may be considered a computer, including magnetic, optical, electrochemical, or other high-speed data processing device or system.

"Computer network" has many definitions in state statutes. Some define "computer network" as "the interconnection of communication lines with a computer through remote terminals or a complex consisting of two or more interconnected computers." One state expressly includes microwave or other electronic communication means for interconnecting computers. Another commonly followed definition of "computer network" is as follows:

> A set of related, remotely connected devices and communication facilities including more than one computer system with capability to transmit data among them through communication facilities.

One state shortens the definition to "an interconnection of two or more computer systems." Some states use the term "computer program," others use the term "computer software," and some use both even though the meaning of the two often overlaps. Nearly half of the states with computer crime statutes use the term "computer software" and define it as "a set of computer programs, procedures and associated documentation concerned with the operation of a computer system." "Computer software" is one of the few terms to be consistently defined in all states that use the term in their computer crime statutes. Unfortunately, "software" as used in computer technology parlance is jargon that has many significantly different definitions.

Another term that has presented definitional difficulties is "data." Only two states use precisely the same language. In every other state, some minor differences appear. One defines "data" simply as information of any kind in any form including computer software. Two others classify data as intellectual property.

In the definition of "financial instrument," the following terms appear in the various state statutes: check, cashier's check, draft, money order, certificate of deposit, letter of credit, bill of exchange, credit card, debit card, marketable security, warrant, note, negotiable instrument, transaction authorization mechanism, and any computer system representation thereof. Most of the items appear in almost all state statutes.

"Property" is still another term that is defined slightly differently in all states. One typical state statute definition of property seems to include most, if not all, of the elements used in different form:

> Any tangible or intangible item of value that includes, but is not limited to, financial instruments, geophysical data or the interpretation of that data, information, computer software, computer programs and computer-produced or stored data, supporting documentation, computer software in either machine or human readable form and any other tangible or intangible item of value.

Another state defines "property" simply as "anything of value" and then lists numerous examples, including "computer programs or data."

These examples illustrate the inconsistencies, inaccuracies, and limitations in the definitions of important terms used in the computer crime statutes of different states. In a few years voice data entry and output, digitized voice, knowledge-based systems, lasers, optics, molecular-based logic, and neural logic could result in new methods and evidence of crime for which current laws might be inadequate.

There are now as many different and conflicting definitions of computer crime as there are states with computer crime statutes. The definitions of those terms, their comprehensibility, rate of obsolescence, and ease of application will play an important role in determining how successfully and effectively these new statutes will be used to deter and prosecute computer crime.

Penalties in State Computer Crime Laws

State legislatures have taken a variety of approaches to providing punishment for computer crime. Generally, the computer crime statute itself does not explicitly state the fine or term of imprisonment to be prescribed to a convicted person. Instead, the penalty provisions classify the computer crime as a particular felony or misdemeanor. To determine the scope of the penalty, one must refer to the general criminal penalty statute of that state. Delaware's computer crime statute illustrates this point. It simply provides that "computer fraud is a Class C felony and computer misuse is a Class E felony." Even more simply, Idaho prescribes a general felony penalty for computer fraud and computer tampering and a misdemeanor penalty for unlawful computer access.

A few computer crime statutes expressly delineate the bounds of the penalties. For instance, the Oklahoma statute sets forth a fine of $5,000 to $10,000 and/or confinement in the state penitentiary for 1 to 10 years for conviction of a felony under the computer crime law. Rhode Island's penalty section provides for a fine of not more than $5,000 and/or imprisonment for not more than 5 years for commission of a computer crime.

Prosecutorial Experience with State Computer Crime Laws

Prosecutors interviewed for the 1985 study reported a number of experiences:

- Only a few of the many incidents investigated resulted in prosecution, primarily because the evidence available did not appear to support indictment. Some prosecutors reported that grand juries failed to understand the case because of the technical nature of the acts involved.
- More perpetrators now seem to be mounting a defense than did those prosecuted in the past. The most actively defended recent cases have been those involving electronic trespass.
- Many prosecutors interviewed were unaware that their state had a computer crime law.
- Some prosecutors reported that because penalties for violation of their computer crime laws are less than under those for traditional theft and burglary laws, they favor use of the more stringent statutes.
- Many prosecutors chose to use the computer crime law only when a traditional fraud, theft, or malicious mischief statute was clearly less applicable.

Prosecutors expressed concern about a number of legal issues related to specific statutes in their state:

- Existing traditional law is not applicable to fraud or larceny by trick when the deceived party is a device (e.g., automated teller machine, vending machine, turnstile, computer). The computer crime statute enacted to address that issue is now considered to be too narrow in focus.
- The definition of "access" taken from the language of an early federal bill that has been adopted by 22 states is too vague. It includes the word "approach," which is appropriate for physical action but not electronic action.
- A recent amendment to the California computer crime law makes electronic trespass without malice a crime and thereby addresses a loophole in many statutes; however, it exempts from the statute such trespass by employees.
- Prosecutors in several states reported a preference for using traditional theft statutes when possible because of their known interpretation by the courts and stronger penalties available.
- One state statute requires victims to report incidents of computer crime to the public authorities. Prosecutors reported that this provision may be unconstitutional.

Timeliness of the Law

The focus of computer crime legislation has been on the current technical aspects of criminal methods and the current technical development of computer products rather than on the more germane but difficult subject of offenses against the information assets at risk, independent of the technology used. The result has been laws that can become quickly obsolete as the technology and its applications change and new technical methods of engaging in information-related crime appear. For example, juvenile hacker attacks on computers demonstrated the absence of laws dealing with electronic criminal trespass into computers. An additional symptom may be the difficulty of producing adequate technical definitions for computer crime laws.

Prosecutors arc inhibited from using computer crime laws by their and the court's lack of computer literacy and the availability of older laws more familiar to them, even though those laws may not be the best or most applicable. Prosecutors report that failure of victims to report suspected computer crimes and to cooperate with prosecution discourages them from developing the capabilities necessary to work in this area of the law. Data communications advances, moreover, have transcended jurisdictional boundaries, causing criminal acts and their effects to fall into different jurisdictions. The concept of geographical proximity is being replaced with electronic proximity as computers become connected to communication circuits. In addition, the perpetration of crimes at remote computer terminals with only electronically produced means of identification of suspects and recording of their activities makes obtaining adequate evidence difficult.

Computer Crime Laws of Selected States

Five state laws are summarized and briefly analyzed below.

Florida Computer Crime Act

Summary: The Florida Computer Crime Act [Fla. Stat. Ann. Sect. 815.01 et seq. (West Supp. 1979)] proscribes several offenses against intellectual property including data and programs, offenses against computer equipment and supplies, and offenses against computer users. Intellectual property includes programs and data existing within or without a computer (system or network). The offenses against intellectual property are willfully and without authority: (1) modifying data, programs or supporting documentation; (2) destroying data, programs, or supporting documentation; and (3) disclosing or taking data, programs, or supporting documentation that are trade secrets or confidential. Such acts are felonies of the third degree unless the offense is committed for the purpose of devising or executing a scheme or artifice to defraud or obtain any property, in which case the crime is a felony in the second degree.

Offenses to computer equipment and supplies (the terms are not further defined by the law) include willfully, knowingly, and without authorization modifying such equipment or supplies. That crime is a misdemeanor of the first degree unless the offense is for the purpose of devising a scheme or artifice to defraud or to obtain any property, in which case the offense is a felony of the third degree. The offense of willfully, knowingly, and without authorization destroying, taking, injuring, or damaging a computer (system, network) or equipment or supplies used or intended to be used in a computer (system, network) is a misdemeanor of the first degree if the damage is $200 or less and a felony of the third degree if the damage is between $200 and $1,000. If the damage is $1,000 or more or if there is an interruption or impairment of governmental operation or public communication, transportation, or supply of water, gas, or other public service, the felony is of the second degree.

Offenses to computer users include willfully, knowingly, and without authorization accessing or causing to be accessed a computer (system, network) or willfully, knowingly, and without authorization causing the denial of computer system services to an authorized user of the services which are owned by, under contract to, or operated for, on behalf of in whole or in part, or in connection with another. The offense is a felony of the third degree unless it is committed for the purpose of devising or executing a scheme or artifice to defraud or obtain property. In that event the offense is a felony of the second degree.

Finally, the law states that it is not intended to preclude the applicability of other Florida criminal law.

Analysis: The law covers acts or theft of and damage to computer equipment, supplies, programs, and data. It covers willful, unauthorized access to computers (systems, networks) and denial of services to users. The offenses to intellectual property (programs and data) apply whether or not the property is stored inside a computer: that is, the law applies to programs and data contained in listings, tapes, disks, cards, and other off-line and on-line media of expression. The law does not require the media of storage to be a "thing," and consequently, electronic impulses should be includable. Such inclusion will ease the finding of a taking when a program is taken, modified, or destroyed over telephone lines, as in the *Ward* [Ward 1972] and *Seidlitz* [Seidlitz 1978] cases.

Because "unauthorized" is not defined by the law and because "access" is defined so poorly, the prohibition against theft of computer services such as computer time is not clearcut. Florida appears to have no specific theft of services statute, and the property theft statute [Fla. Stat. Ann. Sect. 811.021(1)(a) (Supp. 1975)], "anything of value," would have to be interpreted to include services. Because applicability of both the new law and the prior property theft law is unclear, obtaining a conviction for theft of services such as computer time may remain difficult in Florida.

A particular advantage of the Florida law is that computer programs or data stored other than in a computer qualify as intellectual property within the meaning of the statute. This fact will aid in the prosecution of thefts, disclosures, alterations, and destructions that do occur to computer products but were not covered by prior law.

Colorado Computer Crime Law

Summary: The Colorado Computer Crime Law [C.R.S. Sect. 18-5.5-101 (1973, 1978 Repl. Vol.)] proscribes the knowing use of a computer for fraudulent purposes, the assault or malicious destruction of a computer, and the unauthorized use of alteration of a computer or its "software or data. Penalties relate to the value of the item stolen: under $200 of loss or damage is a misdemeanor punishable by a fine and jail sentence up to 12 months; loss or damage over $200 is a felony punishable by a fine and jail sentence up to 40 years.

Offenses that are fraud-related are those in which knowing use ("use is defined to mean to instruct, communicate with, store data in, retrieve data from, or otherwise make use of a computer, computer system, or computer network) is made of a computer (system, network) for the purpose of devising or executing a scheme to defraud; obtaining money, property, or services by false pretenses; or committing theft.

The other form of computer crime is the knowing and unauthorized use, alteration, damage, or destruction of a computer (system, network).

The graduated classification of offense and associated penalties relate to the dollar value of the loss. Currently, these are: under $50 is a Class 3 misdemeanor, $50 to $199 is a Class 2 misdemeanor, $200-$9,999 is a Class 4 felony, and $10,000 and above is a Class 3 felony. (The Class 3 felony also includes offenses, such as child abuse, that result in serious bodily injury.)

Analysis: This legislation is modeled on the Florida law; however, it is narrower in coverage in that data and programs must be "contained in such computer . . ." to be the subject of the Colorado law damage, alteration, or destruction provisions. It also appears that theft or fraud involving property (which includes information and electronically produced data and "software") must be accomplished by use of a computer to fall within the prescriptions of the law.

Further, there is no sanction for denial of computer services unless such denial is part of a scheme to defraud.

The law is in response to the inadequacies of existing law in that it did not contemplate computer abuse and could not be stretched to accommodate the new forms of wrongful activity. In particular, in a case decided by the Colorado Supreme Court sitting *en banc* on March 19, 1979, the court held that the unauthorized reading and later transcription of a medical record without a taking of the physical record did not constitute a theft because the medical information was not a "thing of value" within the meaning of the theft statute. (*People* v. *Home Insurance Co.*, No. 27984.)

The law's definitions—the weak point in most existing and pending computer crime legislation—are somewhat more precise than other attempts in this area, but there are still problems with defining "software" and "hardware" in the dynamic technological milieu.

Arizona Computer Fraud

Summary: The Arizona statute [Ariz. Rev, Stat. Ann. Sect. 13-2301 and Sect. 13-2316 (Swest 1978)] in its general criminal fraud provisions defines in Sect. 13-2301 for the purposes of Sect. 13-2316 various terms with regard to computers—e.g., "access, computer, computer network, computer program, computer software, computer system, financial instrument, property, and services." Section 2316 provides for the offense of computer fraud. This section states that a person commits computer fraud by accessing, altering, damaging, or destroying without authorization any computer, computer system, computer network with the intent to devise or execute any scheme or artifice to defraud, deceive, or control property or services by means of false or fraudulent pretenses, representations, or promises.

Computer fraud in the first degree, punishable by up to 5 years in prison, is committed when a person accesses, alters, damages, or destroys a computer (system, network) without authorization and with intent to devise or execute a scheme to defraud or to control property or services by false or fraudulent pretenses.

Computer fraud in the second degree, punishable by up to 1½ years in prison, is committed by an "unauthorized intentional access, alteration, damage, or destruction of a computer (system, network) or any software, program, or data contained therein."

Analysis: This law, which was passed at about the same time as the Florida law but independent thereof, is similar in that it covers hardware, programs, and services. Note that "software, programs, and data" must be contained in the computer before such "data and programs" are covered by the law. Otherwise, other Arizona law applied to intellectual or intangible property will have to be applied.

The legislature has coined a definition of "software" that encompasses a related group of programs, procedures, and documentation associated with the operation of a computer system. It is of utmost importance when applying any computer crime law to read carefully the definitions therein because they will differ from each other and unfortunately from common usage in the computer field as well.

California Basic Computer Crime Statute

Summary: The California Computer Crime Statute [Calif. Rev. Stat. 1987, Sect. 502, Ch. 1499 (1 January 1988)] has been modified three times in response to advances in technology and computer crime methods and offenses. It covers five offenses: (1) manipulating data, a computer system, or computer network to devise or execute a fraud; (2) knowingly accessing and without permission taking copies or using any data from a computer or taking any supporting documentation, internal or external, to a computer; (3) theft of computer services; (4) knowingly accessing and without permission damaging data, computer software, or computer programs, internal or external, to a computer; and (5) disrupting or denying computer services to an authorized user. The last two offenses cover electronic trespass into a computer, computer system, or computer network.

An infraction of the last two offenses is punishable by a fine not exceeding $250. However, if the victim's expenditure exceeds $5,000, the penalty is a fine not exceeding $5,000 or 1 year in prison. Penalties for the other three offenses are a maximum of $10.000 and up to 3 years in prison. Civil action for compensatory damages is provided. Multiple jurisdictions in which offenses occur are allowed to result in criminal or civil action in any of the jurisdictions.

Computer or computer-related materials may be seized under warrant or arrest and forfeited. (Forfeiture of seized property is pursuant to Section 502.01, which is an error in the statute since the code has no such section.)

Exempted from prosecution is any employee accessing the employer's computer system when acting outside the scope of lawful employment so long as the employee's activities do not cause an injury exceeding $100. The conduct of minors is imputed to the parent or legal guardian.

Analysis: The most recent additions to this law cover the offense of electronic trespass, identification of the victim's expenditure, and provision for confiscation of seized equipment and materials. The losses incurred are identified as the victim's expenditures rather than direct or absolute losses. The victim's expenditures include the efforts necessary to verify that anything was or was not altered, deleted, damaged, or destroyed because of the suspect's access to the victim's computer or computer network. These new provisions are quite innovative and may form the basis of a model for updates to other state laws.

The trespass provision covers hacker intrusion attacks where no other offense may occur other than browsing among data files. This provision is more liberal than the New York State statute that requires due notice be given to a potential intruder in a display screen warning. However, other California statutes may require this warning.

The exemption of employees from prosecution is meant to protect whistle blowers, but it presents a difficult challenge for the prosecutor who must produce strong evidence that the suspected employee did not know that his or her act was not authorized. If the prosecutor cannot prove that the act was not authorized, no offense has occurred. The $100 damage limit by an employee does not specify the period the loss occurs and does not include compensation for the victim's expenditures as part of the penalty.

The technical definitions are similar to those in other state laws. Computer network is more simply defined, however, to mean two or more computer systems connected by telecommunication facilities. Computer program is equivalent to software. The definition of computer system excludes nonprogrammable calculators capable of being used in conjunction with external files, one or more of which contain computer programs, electronic instructions, input data, and output data. This definition will become obsolete rather quickly as calculators increase in capability.

Data may be in any form—in storage media, in transit, or presented on a display device. This definition is an important extension of those in other state statutes because data are in transit for significant periods. Because of the broad definitions of these technical terms, offenses can include acts against data and computer programs in the vicinity of computers.

New York Offenses Involving Computers [NY Book 39, Sect. 156][3]

Summary: New York has several new penal laws designed to meet the problems of computer crime:

- Creation of the new crimes of unauthorized use of a computer [Sect. 156.05] and computer trespass [Sect. 156.10] designed to deal with the unauthorized use of a "computer" or "computer service."
- Creation of the new crimes of computer tampering in the first degree [Sect. 156.25] and in the second degree [Sect. 156.20] to deal with the unauthorized and intentional alteration or destruction of a "computer program" or "computer data."
- Creation of the new crimes of unlawful duplication of computer-related material [Sect. 156.30] and criminal possession of computer-related material [Sect. 156.35], the former designed to deal with the unauthorized duplication of a "computer program" or "computer data," and the latter designed to cover the unauthorized possession of such duplicated material.
- Creation of a new subdivision of the crime of theft of services [Sect. 165.15(10)] to make it a crime for a person who, with intent to avoid payment for the use of a "computer" or "computer service," avoids paying the lawful charge. Further, the term "computer service" is included in the definition of the term "service" as that term is defined for the article involving theft [Sect. 155.00(8)], and would thus include certain thefts of a computer service in such existing crimes as theft of services by a stolen credit card [Sect. 165.15(1)] and unlawful use of credit card [Sect. 165.17].
- Application of the existing crimes of larceny, forgery, false written instruments, and related offenses to such conduct as it relates to a computer program or computer data by inclusion of the terms "computer data" and "computer program" within the definition of the following terms:
  - "Property" as defined for the title involving theft [Sect. 155.00(1)],
  - "Written instrument" as defined for the article involving forgery and related offenses [Sect. 170.00(1)], and
  - "Business record" and "written instrument" as defined for offenses involving false written instruments [Sects. 175.00(2) and (3)].

To ease the sometimes elusive venue of an electronic medium crime, an amendment to the Criminal Procedure Law (20.60) has provided that: "A person who causes by any means the use of a computer or computer service in one jurisdiction from another jurisdiction is deemed to have personally used the computer or computer service in each jurisdiction."

Analysis: The terms "computer," "program," and "data" may have a commonly understood usage. Unlike the generally understood definition of "computer," however, Sect. 156.00 defines it to include various peripherals designed to store, retrieve, or communicate the results of computer operations, programs, or data. The term "computer service" appears to have been devised to refer to the ever-expanding telecommunication industry that supplies information and services via computers (e.g., Westlaw).

The term "computer material" is a list of certain kinds of computer programs and data. The purpose of the term is to single out the listed programs and data for felony treatment when such programs and data are either invaded without authorization [Sect. 156.10(2)] or are altered or destroyed [Sect. 156.20(3)].

A common element of both unauthorized use of a computer and computer trespass is the defined term "uses a computer or computer service without authorization" [defined in Sect. 156.00(6)]. Critically, that term requires *both* that the user lack authorization to use the computer or computer service *and* that actual or specified forms of constructive notice to that effect be given to the user. Proof of constructive notice by showing that the computer was programmed to automatically provide such notice is "presumptive evidence," a permissive inference, that such notice was given.

Effective and provable notice of the lack of authorization to use a computer is further highlighted by the available defense that "the defendant had reasonable grounds to believe that he had authorization to use the computer" [Sect. 156.50(1)]. "Reasonable grounds" imports an objective element in the determination of whether the defendant's belief that he had authorization to use the computer would be one a reasonable person, in the defendant's situation and circumstances, would have. [Cf. *People* v. *Goetz*, 68 N.Y.2d 96 (July 8, 1986).] Noticeably absent as an available defense to the defendant who may have had authorization to use the computer but not a computer service is the claim that the defendant had reasonable grounds to believe that he had authorization to use the computer service.

The second element of the unauthorized use of a computer is proof that the computer had a device or coding system designed to prevent the unauthorized use of the computer or computer service.

The threshold requirements of notice and a system to prevent unauthorized use in order to be held criminally liable for the crime of "unauthorized use of a computer" were deliberately incorporated into the law in order to encourage greater self-protection on the part of the computer industry.

For the computer trespass crime, however, a system need not prevent unauthorized use. Computer trespass requires the notice, and the neither knowingly gaining access to "computer material," defined in Sect. 155.00(5) to mean certain kinds of programs or data listed in the definition, or "an intent to commit or attempt to commit or further the commission of any felony."

No felony need be committed; at a minimum it need only be intended; and the circumstances surrounding the use of the computer or computer service may supply the inference of the requisite intent. [Cf. *People* v. *Mackey*, 1980, 49 N.Y.2d 274, 425 N.Y.S.2d 288, 401 N.E.2d 398.]

Since the crime of unauthorized use of a computer contains an element not contained in the computer trespass crime, it is not a lesser included offense of the computer trespass crime. [See *People* v. *Glover*, 1982, 57 N.Y.2d 61, 453 N.Y.S.2d 660, 439 N.E.2d 376.]

The basic crime, computer tampering in the second degree, requires that a person use a "computer" [defined in Sect. 156.00(1)] or a "computer service" [defined in Sect. 156.00(4)] and without the right to alter or destroy a computer program or data, he intentionally does so. First, a computer or computer service must be the instrumentality of the crime. The unauthorized and intentional destruction of a disk containing a program or data may be criminal mischief, but it is not computer tampering. Second, in addition to showing that the defendant had no right to alter or destroy the program or data, it may be necessary to negate the defense that the "defendant had reasonable grounds to believe that he had the right to alter in any manner or destroy the computer data or the computer program" [Sect. 156.50]. "Reasonable grounds" imports an objective element in the determination of whether the defendant's belief that he had authorization to use the computer would be one a reasonable person, in defendant's situation and circumstances, would have. [Cf. *People* v. *Goetz*, 68 N.Y.2d 96, (July 8, 1986).] Third, the consummated crime requires the actual alteration or destruction of a program or data.

Computer tampering in the first degree initially requires commission of computer tampering in the second degree. Thus, the later crime is a lesser included offense of the former. [See *People* v. *Glover*, 1982, 57 N.Y.2d 61, 453 N.Y.S.2d 660, 439 N.E.2d 376.] The second requirement of the crime is the commission of one of four aggravating elements. The first alternative is that the defendant acted with the intent to commit or attempt to commit or further the commission of a felony. No felony need be committed; at a minimum it need only be intended; and the circumstances surrounding the use of the computer or computer service and the material destroyed may supply the inference of the requisite intent. [Cf. *People* v. *Mackey*, 1980, 49 N.Y.2d 274, 425 N.Y.S.2d 288, 401 N.E.2d 398.]

The second alternative is that the defendant had previously been convicted of a computer offense that is defined in Article 156, or theft of services of a computer or computer service defined in Sect. 165.15(10). Curiously, albeit the law creating these new crimes also expanded the crimes of larceny, forgery, false written instruments, and related offenses to include such conduct as it relates to a computer program or data, a prior conviction of those crimes is not an authorized predicate for the commission of computer tampering in the first degree.

The third alternative is that the computer program or data altered or destroyed be those specifically listed as "computer material" in Sect. 156.00(5). The fourth alternative is that the program or data be altered or destroyed in an amount exceeding $1,000, a sum that parallels the distinction between misdemeanor and felony crimes defined by the value of the property involved.

The new crime of unlawful duplication of computer-related material [Sect. 156.30] is designed to deal with the unauthorized duplication of a "computer program" [defined in Sect. 156.00(1)] or "computer data" [defined in Sect. 156.00(3)]. A related new crime is criminal possession of computer-related material [Sect. 156.35], which is designed to prohibit the unauthorized possession of a program or data duplicated in violation of the crime of unlawful duplication of computer-related material, and possessed with intent to benefit the possessor or a person other than the owner.

The theft of a program or data, through unauthorized duplication, is a crime peculiar to the electronic media. Unlike a traditional larceny, valued and valuable programs or data can be taken without disturbing the rightful owner's possession and without depriving the rightful owner of the program or data. Nevertheless, the program or data can be appropriated by duplication in seconds.

The crime of unlawful duplication of computer-related material requires more than unauthorized duplication. It requires an economic deprivation in excess of $2,500, or irrespective of the amount of economic deprivation, an "intent to commit or attempt to commit or further the commission of any felony." Absent the criminal purpose and absent an economic deprivation of $2,500, the unauthorized duplication of a program or data is not violative of Sect. 156.30.

As with the other computer crimes that make a criminal purpose an aggravating element of the offense, no felony need be committed; at a minimum it need only be intended; and the nature of the program or data duplicated and the circumstances surrounding the duplication may supply the inference of the requisite intent. [Cf. *People* v. *Mackey*, 1980, 49 N.Y.2d 274, 425 N.Y.S.2d 288, 401 N.E.2d 398.]

Finally, in addition to showing that the defendant had no right to duplicate the program or data, it may be necessary to negate the defense that the "defendant had reasonable grounds to believe that he had the right to copy, reproduce, or duplicate in any manner the computer data or the computer program." [Sect. 156.50(3).] "Reasonable grounds" imports an objective element in the determination of whether the defendant's belief that he had authorization to duplicate the program or data would be one a reasonable person, in the defendant's situation and circumstances, would have. [Cf. *People* v. *Goetz*, 68 N.Y.2d 96, (July 8, 1986).]

Other State Authority Bearing on Computer Crime

Automatic Banking Device

Kentucky has a statute [Ky. Rev. Stat. Sect. 434, 685 (Supp. 1978)] that proscribes the misuse of electrical information with regard to automatic banking devices and electronic fund transfers (EFTs). A federal law, Title XX of the Financial Institutions Regulatory and Interest Rate Control Act of 1978 (FIRA), also proscribes EFT crimes.

Credit Card Crime

Many computer crimes consist of or include unauthorized access of a computer system to obtain, alter, damage, or destroy programs, data, or services, such as computer usage. Apart from theft of computer programs (which is discussed separately below), it may be possible to charge a perpetrator with credit card crime, forgery, theft of property, services, or a thing of value under charges of false pretenses and burglary.[4] Most jurisdictions have credit card abuse laws. For example:

- AL Code Tit. 13 Sect. 4-32 4-41 (1977);
- AK Stat. Sect. 11.46 2285 (fraudulent use of credit card), Sect. 11.46.290 (obtaining a credit card by fraudulent means);
- AR Stat. Ann. Sect. 41-2308 (1977);
- GA Code Ann. Sect. 26-1705 to Sect. 26-1705.10;
- HI Rev. Stat. Sect. 851-10;
- IL Ann. Stat. Ch. 121 1/2, Sect. 60, Ch. 121 1/2, Sect. 601 et seq. (Supp. 1978);
- IN Code Ann. Sect. 35-43-51 to 35-43-55 (1979);
- IA Code Ann. Sect. 715.1 to 715.6 (West Supp. 1978);
- KS Crim. Code & Code of Crim. Proc. Sect. 16.841-16.844 (1974);
- KY Rev. Stat. Sect. 434.550-434.730 (Supp. 1978);
- LA Rev. Stat. Ann. Sect. 14.67 (1974);
- ME Rev. Stat. Tit. 17-A Sect. 905 (Supp. 1978);
- MD Crim. Law. Code Ann. Sect. 145 (Supp. 1978);
- MN Stat. Ann. Sect. 609.52 (West Supp. 1979);
- MT Rev. Code Ann. Sect. 94-6-307 (Supp. 1974);
- NV Rev. Stat. Sect. 205.610-205.810 (1977);
- NM Stat. Ann. Sect. 30.16.24-30.16-38 (1978);
- NC Gen. Stat. Sect. 14-113.8-.17 (Supp. 1977);
- OH Rev. Code Ann. Sect. 2913.21 (Supp. 1978);
- SC Code Sect. 16-13-270 and 280 (1976);
- RI Gen. Laws Sect. 11-49-12 to 13 (Supp. 1978);
- SD Compiled Laws Ann. Sect. 22-30 A-8.1 (Supp. 1977);
- UT Code Ann. Sect. 76-6-506.3 (1978);
- WI Stat. Ann. Sect. 943.41 (West Supp. 1979); and
- WA Rev. Code Ann. Sect. 9A.56 (1977).

Whether these may be used to prosecute will depend on the pattern of facts and the statutory language. For example, in some jurisdictions, uttering a fictitious account number is enough to trigger the law. See, for example, Del. Code Ann. Title 11, Sect. 904 (1975) ("credit card" includes writings, numbers, or other evidences of undertaking to pay for property). In other jurisdictions, the actor must actually "utter a fictitious card"; thus, an account number system where no credit cards are actually issued probably would not trigger the statute. See, for example, Va. Code Ann. Sect. 18.1-125.2(2) (Supp. 1974) ("credit card" means instrument or device).

Theft by Deceit

In a Missouri case, *State* v. *Hamma* [569 S.W.2d 289 (Mo. Ct. App. 1978)], decided before the passage of the Financial Institutions Regulatory and Interest Rate Control Act (FIRA), the defendant was accused of stealing by deceit. On appeal he contended that the information did not state conduct constituting the crime charged. The defendant was accused of intentionally stealing $800 by deceit by obtaining someone else's automatic teller bank card and secret identification number and taking money out of the machine at $50 each withdrawal. The defendant contended that he made no representation, let alone a fraudulent representation, and argued that the offense required a verbal misrepresentation to the party defrauded. The court rejected that argument, stating that a misrepresentation could consist of any act, word, symbol, or token calculated and intended to deceive. The court held that the deceit may be made either expressly or by implication. Moreover, the court held that the fraudulent manipulation of an automatic teller is analogous to the use of stolen credit cards, and it cited an earlier D.C. case, *Hymes* v. *U.S.* [260 A.2d 679 (D.C. App. 1970)] as precedent.

In a Virginia case, *Lund* v. *Commonwealth* [217 Va. 688, 232 S.E.2d 745 (1977)], the defendant was charged with theft of keys, computer cards, and computer printouts from a university and using, without authority, computer operation time and services with intent to defraud. The defendant was a graduate student in statistics and a PhD candidate whose dissertation required the use of the computer. He used over $26,000 worth of computer time. The defendant contended that the conviction of grand larceny was faulty because there was no evidence that the articles stolen (e.g., keys, cards, and printouts) were worth over $100 and that computer time and services were not subjects of larceny. The court agreed, holding that the phrase "goods and chattels" could not be interpreted to include computer time and services in view of the rule that criminal statutes must be strictly construed. Moreover, the court held that the unauthorized use of the computer was not the subject of larceny because nowhere in the criminal code section was the word "use" used. The court cited a 1927 case that held that the use of the machinery in spinning facilities did not constitute larceny.

Finally, the Commonwealth contended that although the printouts had no market value, they should be valued by the cost of labor and materials to produce them. The court rejected that argument and also stated that if there was no market value, the only value that could be used was actual value and in this case the only actual value was to the defendant. The court compared *Hancock* v. *State* [402 S.W.2d 906 (Tex. Crim. App. 1966)] (theft of a computer tape containing a valuable program), where the criminal statute was sufficient upon which to base a conviction and the program stolen had a monetary value.

Forgery

To obtain access to another's computer system, the actor will need to discover and use the owner's confidential entry code to the system and account number. The use of this false entry code for the purpose of defrauding or injuring any party may be forgery. Although jurisdictions that have retained the common law requirements of a signature and document would not be applicable, a number of jurisdictions have expanded the common law scope of the crime so that any making, altering, executing, completing, or authenticating of any seal, signature, writing, or symbol of right, privilege, or identification that may defraud or injure another is forgery.

The California Penal Code [Sect. 470 (West 1970)] provides, interalia, that anyone who "... counterfeits or forges the seal or handwriting of another ..." is guilty of forgery. The central question is whether the entry code is either a seal or a signature. The entry code is analogous to the signature on a check (itself a form of computerized draft that uses optical character readers) or the authenticating seal of a notary or official. Moreover, in *People* v. *Burkett* [271 Cal. App. 2d 130, 74 Cal. Rptr. 692 (1969)], the court held that "seal or handwriting" was a "catchall," broad enough to include a photocopy of a reproduction of a seal and a facsimile signature. The defendant had used photocopies of dollar bills in dollar bill changers [271 Cal. App. 2d at 134, 74 Cal. Rptr. at 694].

The New York forgery statute [N.Y. Penal Law Sect. 170.00 et seq. (McKinney 1967)] is a statutory, not common law, offense and covers any false making of private writings that might operate to the prejudice of another.

Delaware, Texas, and Pennsylvania have similar forgery statutes, apparently patterned after the Model Penal Code. Each includes, as protected writings, any symbols of "value, right, privilege, or identification." [Pa. Stat. Ann. Title 18, Sect. 4101(b) (1973); Del. Code Ann. Title 11, Sect. 863 (1975); Tex. Stat. Ann., Penal Code Sect. 32.21(a)(2)(c) (1974).] The offense is a felony in Texas and Delaware and a misdemeanor of the first degree in Pennsylvania.

Thus, at least in some jurisdictions, the use of a false entry code, a symbol of right, privilege, and identification that prints out on any machine and is used to defraud or injure is forgery. As noted in conjunction with credit card abuse, the prosecutor will need to prove a fraud or injury, actual or intended, to trigger the statute. Even though it seems logical that any pecuniary loss should be sufficient, the prosecutor may want to charge at least one of the various theft charges applicable in that proof of value then would not be at issue.

Obliteration or Bugging of Programs

Obliteration or bugging of programs is a form of computer abuse that can be broadly characterized as criminal or malicious mischief. Whereas most jurisdictions have criminal mischief statutes of one type or another that proscribe physical damage to another's personal property, some also have "interference with use" statutes that make it a crime to tamper or interfere with another's property so that the person suffers loss.

Physical Damage

As long as prosecutors successfully characterize the damage, they should have no difficulty when the outward appearance of the disk or tape is unchanged. The problem of successful characterization in California should be minimized by *People* v. *Dolbeer* [214 Cal. App. 2d 619, 29 Cal. Rptr. 573 (1963)]. California's malicious mischief statute, Cal. Penal Code Sect. 394 (West 1970), provides that any malicious injury or destruction of personal property of another is a misdemeanor.

Five other jurisdictions—Massachusetts [Mass. Gen. Laws Ch. 266, Sect. 127 (1968)]; Delaware [Del. Code Ann. Title 11, Sect. 811(a)(1) (1975)]; the District of Columbia [D.C. Code Sect. 22-403 (1967)]; Florida [Fla. Stat. Ann. Sect. 806.13 (Supp. 1976)]; and Virginia [Va. Code Ann. Sect. 18.1-172 (Supp. 1974)]—have malicious or criminal mischief statutes virtually identical to that of California. Penalties generally vary according to the amount of damage (except in Virginia), and large amounts of damage may give rise to felony charges in Delaware and Florida and felony-level punishment in Massachusetts and the District of Columbia.

Unlike the jurisdictions discussed above (which deal with tangible or personal property), New York's criminal mischief statutes use the general word "property" [N.Y. Penal Law Sect. 145.00 et seq. (McKinney 1967)]. But New York [N.Y. Penal Law Sect. 155.00(1)] defines property subject to theft as "money, personal property, or . . . thing in action, evidence of debt or contract, or any article, substance or thing of value." Property for purposes of the criminal mischief and tampering statutes means tangible property. [See R. Denzer and P. McQuillan, Practice Commentary Sect. 145.00, N.Y. Penal Law (McKinney 1967) citing *Polychrome Crop.* v. *Lithotech Corp.* 4 App. Div. 968, 168 N.Y.S.2d 346 (1957) (predecessor to current criminal mischief statute not intended to apply to violations of incorporeal rights).] Thus, although the statute differs slightly from the California statute, the characterization problem is the same.

The New Jersey malicious mischief statutes [N.J. Stat. Ann. Sect. 2A; 122-1 and 17036 (1969)] use differing descriptions of the thing protected; whereas the former refers to personal property, the latter refers to property.

In *State* v. *Shultz* [41 N.J.L.J. 176, 177 (1918)], a lower court emphasized that "in order that the offense of malicious mischief may be perpetrated, it is necessary that there be injury to property; but . . . it is not necessary that the property be entirely destroyed." The operation of the New Jersey malicious mischief statute is unique among all the jurisdictions surveyed. When any malicious mischief occurs, the prosecutor charges a misdemeanor [N.J. Stat. Ann. Sect. 2A; 122-1 (1969)]. But if the prosecutor fails to prove that the value of the property damaged was more than $200, the defendant cannot be convicted of a misdemeanor, but can only be adjudged a disorderly person, punishable by up to 6 months in jail and/or a fine up to $500 [*State* v. *Tonnisen*, 92 N.J. Super. 452, 224 A.2d 21 (1966)].

Pennsylvania's criminal mischief statute is generally inapplicable because Pa. Stat. Ann. Title 18 Sect. 3304(a)(1) and (2) are limited to destruction by dangerous means or so as to cause danger to person or property. However, Subsection (a)(3) appears to incorporate theft by false pretenses and extortion into criminal mischief, perhaps as a smaller included offense of theft. As such, it would be applicable where any loss was caused and the actor used deception to accomplish the mischief. Criminal mischief may be a summary offense, misdemeanor, or felony depending on the amount of loss [Pa. Stat. Ann. Title 18, Sect. 3304(6)].

Two Texas statutes may be relevant in the case of damage to programs. The Texas criminal mischief statute [Tex. Stat. Ann., Penal Code Sect. 28.03 (1947), Subsection (a)(1)] provides that damage or destruction of tangible property of another is an offense. That is not unusual. However, Texas law also proscribes any alteration or destruction of a writing with intent to defraud. While the law resembles the forgery statute in its scope, it extends to any alteration irrespective of what the writing purports to be. [See Tex. Stat. Ann., Penal Code Sect. 32.47 (1974).] Thus, so long as the damage is to printed programs, this provision would be applicable.

The Illinois criminal mischief statute [Ill. Ann. Stat. Ch. 38, 21-1 (Smith-Hurd 1970)] specifically proscribes damage to articles representing trade secrets. The statute provides that knowing damage to property of another is an offense. Property is defined as "anything of value," including articles representing secret scientific information, and this definition applies to all offenses against property. The offense is punishable by up to 5 years in prison and a fine up to $500 if the value of the program damaged exceeds $150.

Interference with Use

Aside from the Pennsylvania statute, which might be used in a tampering situation but does not specifically refer to interference with use as a crime, Pa. Stat. Ann. Title 18, Sect. 3304(a)(3) (1973), statutes in four other jurisdictions make criminal tampering a punishable offense.

Under the general rubric of criminal trespass, the California Penal Code, Sect. 602(j), provides that entry of lands with intent to interfere with any lawful business is a misdemeanor. New York has a broad array of antitampering statutes. N.Y. Penal Law Sect. 145.20 (criminal tampering in the first degree, a Class D felony) would be applicable to any tampering with a publicly owned computer operation. That statute contains a broad provision, Sect. 145.15(1) (criminal tampering in the second degree, a Class B misdemeanor) that applies to any tampering with any property that causes substantial inconvenience. It is also a Class B misdemeanor to create a risk of substantial damage to property whether or not such damage occurs. Substantial damage is defined as damage in excess of $250.

Texas has an analogue to the New York antitampering statute, Tex. Stat. Ann., Penal Code 628.03(a)(2) (1974). A violation is a Class C misdemeanor if the tampering caused substantial inconvenience of no ascertainable monetary amount, and a misdemeanor or felony if the amount of loss is calculable. The Virginia statute [Va. Code Ann. Sect. 18.1-183 (Supp. 1974)] is similar to the California criminal trespass statute discussed above but, unlike the California law, specifically extends its scope to any interference "with the rights of the owner, user, or the occupant thereof . . ." As in California, the offense is a misdemeanor.

Misappropriation of Programs

Computer abuse in this category of misappropriation of programs may take several forms: (a) unauthorized or fraudulent access to programs by an unprivileged user of a facility or by a privileged user of the facility who has no authorized access to the programs; (b) unauthorized or fraudulent disclosure of proprietary programs by an employee, former employee, or contract program developer. The leading reported case of this category is *Hancock* v. *State*, 1 CLSR 562, 402 S.W.2d 906 (Tex. Crim. App. 1966). In *Hancock*, the defendant-employee offered a listing of 59 programs for sale to a person he thought was an agent of one of his employer's clients.

The scope of state criminal laws protecting programs is often determined by whether the programs are included with property otherwise subject to protection. An initial question is whether unpatented and uncopyrighted programs may be protected by criminal trade secret laws. In states that have no trade secret laws, or where dual charges of larceny and theft of trade secrets may be maintained [see, for example, *Ward* v. *Superior Court*, 3 CLSR 206 (Memorandum opinion 51629, 1972)], the prosecutor must determine whether programs are property subject to larceny. In states that have no criminal trade secret laws, the prosecutor must often look to general "offenses against property" statutes to punish the type of computer abuses noted above. Such general statutes are almost the exclusive remedy in all states for obliteration or bugging.

Computer programs, a form of intangible intellectual property, should be protected by state criminal laws. For excellent discussions of the inadequacy of civil remedies, see Comment, Industrial Espionage: Piracy of Secret Scientific and Technical Information, 14 U.C.L.A. L. Rev. 911, 927 (1967), and Comment, Protection of Trade Secrets in Florida, 24 U. Fla. L. Rev. 721 (1972). First, without protection, a program developer has little incentive for creating and investing. Second, it is only just that laborers enjoy the fruit of their labors. Third, the criminal law must prevent misappropriation, misuse, and distortion of proprietary programs. See Galbi, Copyright and Unfair Competition, 3 CLS Sect. 4-3, Art. 1, and Bender, Trade Secret Protection of Software, 38 Geo. Wash. L. Rev. 51629 (1972).

With the exception of trade secrets laws, almost all state offenses against property statutes antedate the advent of computers. Definitions and case interpretations may make prosecution for abuse of an intangible difficult. For instance, abuse of programs by copying or unauthorized communication may be seen as a mere disclosure of an idea. Malicious mischief may be deemed only a rearrangement of magnetic discontinuities with no requisite damage or destruction to the tangible property carrying the programs.

Whether a particular abuse may be successfully prosecuted under larceny or malicious mischief statutes may turn on the skill of the prosecuting attorney in framing the charge where a person has misappropriated programs contained on a magnetic tape or on a printout. *Hancock*, 1 CLSR 562, 402 S.W.2d 906 (Tex. Crim. App. 1966), shows that so long as the value of the intangible intellectual property is added to the value of the tape or paper (a reasonable addition in that it is doubtful that the tape or paper would have been stolen but for the program value) an indictment or information charging grand larceny should be upheld against a motion to dismiss. A closer question might concern the actor who was ignorant of the program's existence but set out to steal bulk paper or computer tapes per se. The general rule appears to be that the prosecution is not required to prove knowledge of value by the thief [see, for example, *People* v. *Earle*, 222 Cal. App. 2d 476, 35 Cal. Rptr. 265 (1963)], and that the market value is fair market value to disinterested buyers and sellers. See also *People* v. *Dolbeer*, 214 Cal. App. 2d 619, 623, 29 Cal. Rptr. 573, 575 (1963) (the value of telephone company customer lists is determined by "effort . . . efficiency . . . and . . . secrecy . . .," not the paper alone).

Where an actor obliterates or bugs programs by altering the magnetic tape or printout, the prosecutor must urge that the "property" that was "injured" under the common form of malicious mischief statutes was the tangible tape or paper. What gives the paper or tape value is the program [see *Hancock* v. *Decker*, 1 CLSR 858, 379 F.2d 552 (5th Cir. 1967)]; when one obliterates the program he obviously injures the tape by rendering it unfit for its purpose. Just as in larceny prosecutions, the prosecutor must be careful to characterize the conduct so as to bring it within the statutory proscription, for example, (1) the thing injured was a tangible tape, and (2) the injury was the obliteration of the program. This method of characterization was suggested by John Kaplan, former prosecuting attorney, currently Professor of Law, Stanford School of Law.

Only in two instances will the abuse of programs probably be unprotected under common larceny statutes. First, where the actor copies a program on his own paper or tape and asports the copies but leaves the originals, he has not committed common law larceny as interpreted in most jurisdictions. But see *Ward* v. *Superior Court*, 3 CLSR 206 (Memorandum opinion 51629, 1972, sustaining a grand theft charge under a similar fact pattern). The result in *Ward* is logical, since one who asports a copy of a program steals both value and control of the property. But the fact that so many states have found a need specifically to proscribe copying a trade secret, Cal. Penal Code Sect. 499c(b)(3)-(4) (West 1970), demonstrates how resistant most courts have been to accepting value or control theories as equivalent to the more traditional "permanent deprivation" theory of larcenous intent.

Second, when a person takes knowledge or electronic signals, he has probably not committed larceny within common law statutes. In *Ward* v. *Superior Court* [Ward 1973], Judge Sparrow stated that electronic impulses ". . . are not tangible and hence do not constitute an 'article' capable of being stolen within California's trade secrets law" [3 CLSR 206, 208 (Memorandum opinion 51629, 1972)]. This opinion may well represent the popular perception of electronic impulses as outside the scope of property protected by statute. As to theft of knowledge, theses that ideas may not be stolen seems to preclude prosecution of those who develop a program and use the knowledge gained thereby for a competitor or for themselves. But see Tex. Stat. Ann., Penal Code Sect. 31.05(b)(3) (1974) (any communication or transmission of a trade secret without consent is a felony of the third degree).

When actors misappropriate computer programs stored in a computer, they may run afoul of several other types of laws. First, the state may denominate misappropriation of trade secrets as a separate and distinct offense. Second, notwithstanding trade secrets laws, the actors may be guilty of larceny; as a corollary, the recipients of the program, other than the actors, would be receiving stolen goods. Third, the offender may have committed one or more of the crimes set forth above.

Trade Secrets

The Restatement test of a trade secret is that the process, item, etc., be used in the trade or business, be kept secret, and give the owner a competitive advantage over those who do not know it. Trade secret misappropriation statutes are enormously useful in cases of program theft but should be analyzed carefully to make sure the technical requisites have been met. [For example, in *Ward* (1972), the judge held that the transference of electronic impulses did not constitute a taking.]

Larceny statutes are relevant in three different contexts related to trade secrets. First, in states that have misappropriation of trade secrets as a separate and distinct offense, a dual charge of larceny and theft (or abuse) of trade secrets may arise from the same act [cf. *Ward* v. *Superior Court*, 3 CLSR 206 (Memorandum opinion 51629, 1972)]. This does not mean, however, that double punishment may be meted out when an actor engages in a single, indivisible transaction that may encompass several crimes. Only the single, heaviest punishment of all the crimes may be imposed. The critical question is what constitutes a single indivisible transaction. Second, where theft of trade secrets is subsumed into the general larceny statute, the burden of the prosecutor to prove trade secrets as property subject to larceny is eliminated. Third, even where trade secrets have not been statutorily included as property subject to larceny, the prosecutor may be able to prove that the secret is a "thing of value."

The New York larceny statute, N.Y. Penal Law Sect. 155.30 (McKinney Supp. 1974), is an excellent example of how a jurisdiction may include trade secrets, "secret scientific material," in its larceny statute. Both stealing and copying are separate offenses, each a Class E felony. If the trade secret has a readily ascertainable value (market or replacement value, see Sect. 155-20) in excess of $1,500, the prosecutor may desire to waive prosecution under Sect. 155.30 and instead charge second degree grand larceny, Sect. 155.35, a Class D felony punishable by 1 to 7 years in prison and a discretionary fine similar to that for Class E felonies.

Unlike New York law, the California theft statute, Cal. Penal Code Sect. 48a (West 1970), nowhere specifically includes trade secrets as property subject to theft. Whereas the trade secret provision, Cal. Penal Code Sect. 499c (West 1970), is probably the exclusive sanction for copying a trade secret without asportation, [cf. *Bender*, Trade Secret Protection of Software], the *Ward* [1972] case indicates that a dual charge of theft and theft of trade secrets is maintainable where an article representing a trade secret, or a copy thereof, is asported.

Although New York is the only state that has incorporated trade secrets into both its own and a general larceny statute, at least three states (Pennsylvania, Massachusetts, and Illinois) have incorporated trade secret protection into theft or larceny statutes without denominating abuse of trade secrets as a separate offense from theft or larceny generally. Ordinarily, trade secret protection can be incorporated into theft or larceny statutes in three ways: (1) consolidation of theft of trade secrets into a theft or larceny statute, as in Pennsylvania; (2) definition of trade secret theft as larceny, as in Massachusetts; or (3) including of trade secrets in lists or property protected by larceny statutes, as in Illinois.

Privacy Invasions

Almost every state has one or more statutes proscribing invasions of privacy by persons in the public sector. Bills, pending in some states, may affect the private sector as well. Some of these statutes carry criminal penalties that may be invoked when an unauthorized and willful disclosure of personal information is made from a computer database.

Federal Penal Laws

The most important federal laws for prosecuting computer crime and computer-related crime are the Communications Fraud and Abuse Act of 1986, the Electronic Communications Privacy Act of 1986, the Credit Card Fraud Act of 1984, the Federal Copyright Act of 1976, and the Wire Fraud Act. These laws are briefly analyzed below.

Computer Fraud and Abuse Act of 1986

The Computer Fraud and Abuse Act of 1986 [18 U.S.C., Ch. 47, Sect. 2101-2103, Sect. 1030 Fraud and Related Activity in Connection with Computers] is the result of many tortuous years of producing draft bills in both houses of Congress. The final adopted version, the result of a compromise between the U.S. Department of Justice and two House of Representatives committees, covers only limited forms of computer abuse. The U.S. Secret Service, in addition to the FBI, has been given explicit jurisdiction to investigate the six offenses specified in the statute, which involve "knowingly and intentionally access[ing] a computer without or exceeding authorization and thereby."

1. Obtaining restricted military foreign relations or atomic energy information to be used to injure the United States.

   Penalty: Fine and/or 10 years in prison (20 years if repeated offense).

2. Obtaining information in financial or consumer credit records from financial or credit-reporting institutions.

   Penalty: Fine and/or 1 year in prison (10 years if repeated offense).

3. Affecting the use of a U.S. government or contractor computer.

   Penalty: Fine and/or 1 year in prison (10 years if a repeated offense).

4. Furthering a fraud or obtaining anything of value excluding usage in a financial institution or U.S. government computer.

   Penalty: Fine and/or 5 years in prison (10 years if repeated offense).

5. Altering, damaging, using, or destroying information in or prevents authorized use of a financial institution or U.S. government computer causing more than $1,000 loss or loss in personal medical care.

   Penalty: Fine and/or 5 years in prison (10 years if repeated offense).

6. Trafficking in any password or similar information for unauthorized computer access if it affects interstate or foreign commerce or the U.S. government.

   Penalty: Fine and/or 1 year in prison (10 years if a repeated offense).

By a memorandum of understanding, the FBI currently has jurisdiction over cases involving national security, terrorism, banking, and organized crime. The Secret Service has jurisdiction over all other cases.

Accessing a computer is defined only in terms of exceeding authorized access; otherwise, accessing is not defined and could mean merely dialing a telephone number assigned to the input port of a dataswitch or computer. The definition might also require a computer acknowledgment that facilitates the use of the computer by the person authorized to use it.

"Without or exceeding authorization" puts the onus of defining an offense on the owner or management of a computer. Therefore, an investigator or prosecutor must obtain information on exactly what is authorized and not authorized in a particular computer environment. For example, policy concerning personal use of an employer's computers determines whether using the computer for other than employer-specified purposes is an offense.

The definitions of the six offenses have raised several questions. Offense #3 involving use of a computer is so wide-ranging that materiality may be questioned. Offense #5 poses the problem of identifying the monetary loss; that could involve determining the computer usage rates, as well as the consequential losses associated with the act (e.g., lost staff time, replacement cost of destroyed or damaged information). Offense #6 is aimed in part at electronic bulletin board system operators and users; questions may arise about user IDs that may be public in one system but confidential in another.

The statute contains only one technical definition. A computer is defined as "an electronic, magnetic, optical, electrochemical, or other high-speed data processing device performing logical, arithmetic, or storage functions and includes any data storage facility or communications facility directly related to or operating in conjunction with such device, but such term does not include an automated typewriter or typesetter, a portable hand-held calculator or similar device." The use of the terms "computer program", "software", "computer system", and "computer network" often found defined in state laws are absent from the federal statute.

This definition of computer may cause considerable difficulties for prosecutors, especially as the technology advances and changes. Data storage and communications facilities probably mean connected equipment rather than physical facilities such as a computer media vault. The exclusion of automated typewriters or typesetters is particularly vexing; presumably, these exclusions were meant to restrict application of the statute to the publishing industry. Excluding a portable hand-held calculator may be appropriate today, but in the near future portable hand-held calculators will probably be more powerful than today's microcomputers. Technical definitions generally tend to make criminal statutes obsolete.

Electronic Communications Privacy Act of 1986 (18 U.S.C.; Ch. 65 Sect. 1367; Ch. 119, Sect. 2510-2521; Ch. 121, Sect. 2701-2710; Ch. 206, Sect. 3121-3126)

The 1986 Electronic Privacy Act extends the current privacy guarantees for traditional telephones to communications involving cellular phones that operate by high-frequency radio waves, transmissions by private satellite, paging devices, and messages transmitted and stored in computers, known as "electronic mail." The law expands and updates 1968 legislation (PL 90-351) that specifies when and how the government can wiretap conventional telephones, but says nothing about new forms of communication that use more modern forms of technology to transmit messages.

While Justice Department officials realized there were gaps in the law, they were initially reluctant to tamper with the statute for fear of weakening the department's ability to use wiretaps as a law enforcement tool. But the electronics industry helped convince Justice that the law had to be amended to deal with technological change and to make their products more marketable.

The basic premise behind the legislation is to protect the content of private communications, regardless of the means of transmission. The law includes the following major provisions:

Definitions and Exemptions

- Rewrote the 1968 wiretap law to protect "electronic communications" and "electronic communications system."
- Defined electronic communications to include "any transfer of signs, signals, writing, images, sounds, data or intelligence of any nature that is transmitted in whole or in part by wire, radio, electromagnetic, photoelectronic or photo-optical system that affects interstate or foreign commerce."
- Defined electronic communications system to mean any wire, radio, electromagnetic, photo-optical or photoelectronic facilities for the transmission of electronic communications, and any computer facilities or related electronic equipment for the electronic storage of such communications.
- Exempted from coverage—and thus left unprotected from intrusion—any radio communication that is "readily accessible to the general public."

Also exempted were the radio portion of a cordless telephone communication, which is transmitted between the cordless telephone handset and the base unit, any communication made through a tone-only paging device, communications between amateur radio operators, general mobile radio services, marine and aeronautical communications systems, police, fire, civil defense and other public safety radio communications systems, and specified satellite transmissions.

- Protected radio signals in several instances: if the signal were scrambled or put into code, that is "encrypted"; if the signal's frequency were changed to one withheld from general use by the Federal Communications Commission (FCC); if the signal were transmitted through a common carrier, like a cellular telephone company that serves the public; or if the signal were transmitted via specific radio frequencies set out in the bill.

Private Interception

- Made it illegal for individuals to intercept electronic communications as defined in the bill.
- Made the offense a felony with a penalty of a fine, a prison term of up to 5 years or both when the interception is for any illegal purpose, such as gathering stock information for insider trading, is for direct or indirect commercial gain, or is an interception of a scrambled or encrypted signal.

Generally, when the interception is of a radio communication, the penalty would be a maximum prison term of 1 year and/or a fine. There are three exceptions: when the interception involves the radio portion of a cellular phone call, specified mobile radio services, or a paging service. In these instances, the violation would bring only a $500 criminal fine.

- Required the government to prove that a defendant intentionally sought to intercept protected communications.
- Established a reduced penalty structure for individuals who intercept satellite transmissions for private use. A first offender would be fined $500 and the government could sue to halt the interception. If an injunction were granted. the judge could then use several means to enforce the order, including citing the defendant for civil or criminal contempt for failure to obey the order.
- Provided a $500 fine for the second and any subsequent offense.
- Authorized the person whose communication was intercepted to sue the alleged perpetrator in federal court.
- Authorized a damage award of the greater of actual damages or statutory damages from $50 to $500 on the first offense, and of actual damages or statutory damages between $100 and $1,000 on the second offense.
- Provided for stiffer penalties when the interception was for commercial advantage or illegal purpose, such as to intercept stock information for insider trading. The violator could be ordered to pay the greatest of $10,000, $100 for each day of the violation or the sum of actual damages suffered by the plaintiff and any profits made as a result of the violation.
- Made it illegal for a person or entity providing wire or electronic communications service to the public to divulge knowingly the contents of any communication except to the person sending the information or to the intended recipient.
- Allowed disclosure of a communication with the consent of the originator or recipient to a law enforcement agency when it appears a crime has been committed.

Government Interception

- Allowed the government to intercept electronic communications after obtaining a court order. Judges could grant the order after they had determined that the interception "may provide or has provided" evidence of any federal felony.
- Allowed law enforcement officials to get court approval for a "mobile tracking device" that goes beyond the geographic jurisdiction of the court. The only proviso is that the device, which is used to track a moving suspect, be installed in the jurisdiction of the judge who approved the order. (Federal courts are divided by geographic region within the 50 states and selected federal territories.)
- Expanded from the 1968 law the list of officials who can seek a court order and expanded the number of crimes for which an interception is authorized.
- Gave the government the right, in limited circumstances, to obtain a roving telephone tap that would enable tapping several phones. Under current law, the government is required to specify which phone officials are going to tap. The government would have to explain why specifying a particular phone "is not practical."
- Made it a felony for any person to divulge information about a possible communication interception by the government in order to obstruct, impede, or prevent such interception.

Stored Communications

- Protected the privacy of stored communications, either before or after delivery if a copy is kept.
- Made it a misdemeanor to break into any electronic system holding copies of messages either before or after delivery or to exceed authorized access in the system to alter or obtain the stored messages.
- Provided a fine for a first offense of up to $250,000, a maximum 1-year prison term or both, if the offense were committed for commercial advantage of "malicious destruction or damage." There would be a 2-year prison term for a second offense.
- Provided a maximum fine of $5,000 or imprisonment of up to 6 months for an offense that was not for commercial gain or for malicious destruction or damage.

- Allowed the government to require disclosure of copies of electronic mail, no matter how long it has been stored, if the government obtained a warrant.
- Gave the government the additional option of using a subpoena to get information when the information has been stored more than 6 months. To do that, it must give prior notice to the electronic-mail customer involved. In addition, the subpoena must be issued by a government agency authorized by statute to do so or by an agency acting at the request of a grand jury.
- Required that the court order shall be issued only if the governmental entity shows that there is reason to believe the contents of the electronic communication or the records are "relevant to legitimate law enforcement inquiry."
- Gave the electronic-mail customer the right to file a motion to quash any subpoena or vacate any court order. The person seeking to block the subpoena or court order would have to show that the records sought "are not relevant to a legitimate law enforcement inquiry" or that the government had not complied with the procedures established for obtaining information.

Pen Registers; Trap and Trace

- Established standards for government use of "pen registers" and "trap and trace devices."
- Defined a pen register as a device that records or decodes numbers dialed or otherwise transmitted by telephone. However, devices used to monitor calls for billing or recording incident to billing are not covered.
- Defined a trap and trace device as one that captures an incoming electronic or other impulse and can identify the number from which a call was made.
- Barred generally the use of pen registers and trap and trace devices except pursuant to a court order.
- Provided a penalty for knowingly violating this section of a fine and imprisonment of up to one year or both.
- Required a government agency seeking a court order to use a pen register or trap and trace device to certify that the information "likely to be obtained is relevant to an ongoing criminal investigation being conducted by the applying government agency."
- Specified the elements a court-order granting approval for a pen register or trap and trace device must contain. The order must include the identity of the person whose phone will have the pen register or trap and trace device attached to it, the identity of the person who is the subject of the investigation, the number and physical locations of the telephone lines to be monitored and a statement of the offense to which the information obtained is expected to release.
- Limited the use of the device to 60 days, although 60-day extensions could be granted upon application.
- Specified that the order approving installation of a pen register or a trap and trace device shall be sealed until otherwise ordered by the court.
- Required that the provider of the wire or electronic communications service to be monitored cooperate with law enforcement agencies in installing the pen register or trap and trace device.
- Required that the provider of such service be "reasonably compensated for reasonable expenses" for his cooperation and facilities.

The Credit Card Fraud Act of 1984 (18 U.S.C., Ch. 47, Sect. 1029)

This law was enacted primarily in the interest of controlling credit card fraud and card counterfeiting. Because the definition of credit card is broad, the law covers many computer-related offenses. The objects of the offenses are "access devices," which are defined as "any card, plate, code, account number, or other means of account access that can be used alone or in conjunction with another access device, to obtain money, goods, services, or any other thing of value or that can be used solely to initiate a transfer of funds (other than a transfer originated solely by paper instrument)." Therefore, the following access devices could be included as the object or tool of an offense:

Magnetic stripe credit card	Computer user ID
Smart card	Computer password phrase
Credit plate (containing an address)	Computer password
Dynamic password card	Cryptographic key
Key	Computer account number
Any shaped device for machine insertion	Answers to a question or series of questions
Account balance or state of account	Banking account number
Computer access protocol data	PIN (personal identification number)
Unlisted telephone number	PAN (personal account number)

The exclusion of devices associated with paper instruments in funds transfer would presumably exclude document seals, wax seals, and stamps. Any knowable, but private information used for access to computers seems to be included.

In a computer crime context, the offenses include the following:

1. Fraudulently producing, using, or trafficking in one or more counterfeit devices.
2. Fraudulently trafficking in or using one or more unauthorized devices and obtaining anything of value aggregating $1,000 or more during a one-year period.
3. Fraudulently possessing 15 or more counterfeit or unauthorized devices.
4. Fraudulently producing, trafficking in, having control or custody of, or possessing device-making equipment.

Punishment ranges from $10,000 to $100,000 or 10-20 years in prison. Multiple convictions increase penalties.

This law may be used against deceitful hackers and pirate bulletin board system operators who search for, test, and exchange computer access telephone numbers, user IDs, passwords, and access protocols for computer intrusion. The law is also applicable to a wide range of computer intrusion activities in the federal jurisdiction.

Federal Copyright Act of 1976

Theft of computer programs can be prosecuted under federal copyright laws. The copyright office has accepted registration of computer programs as "books" since 1964. The House Committee Report on the Copyright Act of 1976, p. 54, states that the term "literary works . . . includes computer data bases, and computer programs to the extent that they incorporate authorship in the programmer's expression of original ideas, as distinguished from the ideas themselves." Thus authors of computer programs can protect documentation and lines of computer code from copying, but copyright protection does not extend to programmers' algorithms.

In addition to providing for civil actions and damages for copyright infringement, the Copyright Act of 1976 also provides for criminal penalties for infringement and for fraudulent removal of copyright notice. Criminal liability for infringement is proven by showing the elements of civil infringement, ownership in another party and copying, and, in addition, by demonstrating willfulness and financial gain [17 U.S.C. Sect. 506(2)]. Section 506(a) provides for a maximum penalty of 1-year imprisonment and a $10,000 fine. Section 506(b) also provides for a mandatory forfeiture and destruction of all infringing copies.

Section 506(d) makes it a criminal act to remove or alter any notice of copyright with a fraudulent intent. This conduct is criminal even though it creates no civil liability. Anyone convicted of such an act is subject to a $2,500 fine.

State laws purporting to describe criminal or lawful conduct involving copyright infringement under federal laws are invalid under the doctrine of federal preemption.

Wire Fraud Act (18 U.S.C. Sect. 1343)

The elements of Sect. 1343 are identical to Sect. 1341, with the exception of the federal medium abused. When one uses a remote terminal to perpetuate a computer fraud, or when one telephones an accomplice, so long as the "message" crosses state lines, the statute is applicable. All reported cases involving Sect. 1343 have dealt with conversations that crossed state lines, leading one to believe that the message must, in fact, cross state lines. Since Sect. 1343 does not use the word "facility," jurisdiction hinges on use of an interstate wire, notwithstanding the fact that "[i]t cannot be questioned that the nation's vast network of telephone lines constitute interstate commerce" [*United States* v. *Holder*, 302 F. Supp. 296, 298 (D. Mont. 1969)]. It is not clear that the use of the word "facility" in any new legislation would embrace interstate calls either; see *United States* v.

DeSapio, 299 F. Supp. 436, 448 (S.D.N.Y. 1969) (construing phrase "facility in . . . interstate commerce" as requiring interstate calls for 18 U.S.C. Sect. 1952), because there may be a distinct difference between facilities "in" interstate commerce and facilities "of" interstate commerce. Both mail fraud and wire fraud are very useful aids to the prosecution of computer crime.

Other Federal Authority Bearing on Computer Crime

Financial Institutions Regulatory and Interest Rate Control Act of 1978 (FIRA)

Title XX of FIRA, the Electronic Fund Transfer Act, is designed to define and provide for individual consumer rights as they are affected by electronic funds transfer (EFT). In so doing, the act provides federal regulation of EFT by establishing the rights, liabilities, and responsibilities of participants, including financial institutions, consumers, and other users of EFT.

Section 916 of the act is the criminal liability section, which most directly concerns or at least may have a bearing on computer crime in the federal arena. It provides for a fine of not more than $5,000, imprisonment for not more than 1 year, or both for anyone who knowingly and willfully gives false information or fails to provide information required by the act or regulations promulgated thereunder, or otherwise fails to comply with the act or its regulations.

The second section of the criminal liability provision imposes a fine of not more than $10,000, imprisonment for not more than 10 years, or both for the following six acts when interstate or foreign commerce is involved, when the money, goods, services, or things of value involved have a value of $1,000 or more when aggregated over a 1-year period and when a counterfeit, fictitious, altered, forged, lost, stolen, or fraudulently obtained debit instrument is involved. The term "debit instrument" means a card, code, or other device by which a person may initiate an EFT. The six acts include:

- Knowingly using or attempting or conspiring to use a debit instrument, as described above, to obtain anything of value, as described above.
- With unlawful or fraudulent intent, transporting or attempting or conspiring to transport a debit instrument knowing that it is counterfeit, stolen, etc.
- With unlawful or fraudulent intent, using an instrumentality of interstate or foreign commerce to sell or transport a debit instrument knowing it is counterfeit, stolen, etc.
- knowingly receiving, concealing, using, or transporting anything of value (except tickets for interstate or foreign transportation) which has moved in interstate or foreign commerce and has been obtained with a counterfeit, stolen, etc., debit instrument.
- Knowingly receiving, concealing, using, selling, or transporting one or more tickets for interstate or foreign transportation whose value aggregated within a 1-year period is $500 or more and was obtained or purchased by means of a debt instrument that was counterfeit, stolen, etc.
- In a transaction affecting interstate or foreign commerce, furnishing anything of value through the use of a counterfeit, stolen, etc., debit instrument knowing that it is counterfeit, stolen, etc.

The Federal Privacy Act of 1974

The Privacy Act of 1974 is codified in 5 U.S.C. Sect. 552a. The criminal penalties for violation of its provisions are contained in Subsection (i)(1)-(3). These criminal penalties may be invoked when a violation of the act, resulting from an unauthorized and willful disclosure of personal information, is made from a computer data base.

The basic provisions of the act are to protect the privacy of individuals. Therefore, an agency, as defined in 5 U.S.C. Sect. 551(1) and 552(e), is prohibited, with a variety of exceptions, from disclosing any record contained in a system of records to anyone or another agency unless the individual has made a written request or has given prior written consent.

If any officer or employee of an agency, knowing that disclosure of specific material is prohibited either by the act or regulations promulgated thereunder, willfully discloses the material to a person or agency not entitled to it, the officer or employee has committed a misdemeanor and will be fined not more than $5,000.

The same penalty is applicable to an officer or employee who willfully maintains a system of records, which could include a computer data base, without complying with the notice requirements of Subsection (e)(4). Subsection (e)(4) requires each agency that maintains a system of records to publish in the *Federal Register* not less than once a year a notice of the existence and character of the record system. The notice must include the system's name and location, the categories of individuals, records and their sources included, the routine use of the records, the system's storage, retrieval, access control and disposal policies and practices, the responsible agency official, procedures used to notify individuals, at their request, that records are contained regarding that individual and procedures for an individual to gain access to the records and to contest the contents of the records.

Finally, the same criminal penalties are applicable to anyone who knowingly and willfully requests or obtains under false pretenses a record regarding an individual.

Federal Criminal Code Provisions

At least 40 sections of Title 18 of the United States Code bear directly or indirectly on computer abuse. For ease of analysis, these are grouped into seven broad categories: theft and related offenses; abuse of federal channels of communication; national security offenses; trespass and burglary; deceptive practices; property damage; and miscellaneous.

Theft and Related Offenses

18 U.S.C. Sect. 641 (Embezzlement or Theft of Public Money, Property, or Records): The basic statute that protects federal property from theft is 18 U.S.C. Sect. 641. The statute covers both the thief and the receiver of stolen property. Although most of the terms of the statute are straightforward, several bear directly on computer abuse because of their expansive meanings.

One who "knowingly converts" public property violates Sect. 641. It is no defense to a charge of unlawful conversion that one intended to return the property [cf. *Morissette* v. *United States*, 342 U.S. 246 (1952)]. "[C]onversion . . . may be consummated without any intent to keep . . ." 342 at 271-272, or make restitution, unless those acts negate the requisite mens rea. While no court has ever considered whether one may "embezzle," "steal," or "purloin" programs by unprivileged copying or otherwise, it is highly likely that any unprivileged abuse may be styled a "conversion."

> Conversion, however, may be consummated without any intent to keep and without any wrongful taking, where the initial possession by the converter was entirely lawful. Conversion may include misuse or abuse of property. It may reach use in an unauthorized manner . . . It is not difficult to think of intentional and knowing abuses and unauthorized uses of government property that might be knowing conversions but which could not be reached as embezzlement, stealing or purloining. Knowing conversion adds significantly to the range of protection of government property . . . 342 U.S. at 271-272. See also *United States* vs. *Tijerina*, 407 F.2d 349 (1-th Cir. 1969), cert. den. 396 U.S. 843 (1969) (deprivation of control of trucks for a period of time an unlawful conversion within Sect. 641).

The notion of "conversion" is as broad as the definition of the res that is public property. Moreover, the statute itself is broad enough to include theft of labor or services, *Burnett* v. *United States*, 222 F.2d 426 (6th Cir. 1955) (wrongful conversion of services and labor to two army servicemen by army officer), and uses the catchall phrase "any . . . thing of value . . ."

The meaning of the phrase "of the United States or of any department or agency thereof" is broader than absolute ownership. An agency of the United States is, among other things, "any corporation in which the United States has a proprietary interest . . .;" 18 U.S.C. Sect. 6. "Proprietary interest" is broad enough to include any ownership of stock. Cf. *United States* v. *Anderson*, 45 F. Supp. 943, 946 (S.D. Cal. 1941) (discussing predecessor to Sect. 641). It may be enough if the United States has the power to control the use of the res, *Bernhardt* v. *United States*, 169 F.2d 983 (6th Cir. 1948) (property under Army control at Army depot protected by Sect. 641), even if the res is in private hands. *United States* v. *Echevarria*, 262 F. Supp. 373 (D.P.R. 1967) (advances of United States funds paid to university are protected by Sect. 641).

Although there are no cases directly on point, it seems clear that a joint interest, divided or undivided, or an equitable interest, such as a right to use, may be converted. Thus, should the government purchase the right to use certain programs, and those programs be misappropriated, prosecution should be available under Sect. 641. In addition, one case suggests that property in government custody or possession, even if the government has no legal or equitable title thereto, may be the subject of theft. See *United States* v. *Gardner*, 42 F. 829 (N.D. N.Y. 1890) (custom booty awaiting foreclosure as ressubject to theft).

It is clear that if programs are being developed for the government, their theft or conversion violates Sect. 641. Moreover, *United States* v. *Anderson* shows that raw materials may well be included under this clause [45 F. Supp. at 945-949].

In its broadest interpretation, any misappropriation of programs that are subject to some measure of government control, custody, or ownership is a violation of Sect. 641.

At least two decisions have dealt with 18 U.S.C. Sect. 641. *United States* v. *Digilio*, 538 F.2d 972 (3d Cir. 1976), was a conviction for conspiracy to defraud the United States and to convert to the defendant's own use the records of the United States, particularly photocopies of official files of the FBI. Defendants contended that Sect. 641 was inapplicable because the government was not deprived of the use of the information contained in the records. They contended that the unauthorized copies of government records were not themselves records and that the unauthorized transmission of the information is not proscribed by Sect. 641.

The government had based its argument of Sect. 641 applicability on *United States* v. *Bottone*, 365 F.2d 389 (2d Cir. 1966) cert. den. 385 U.S. 974, 87 S. Ct. 514, 17 L. Ed. 2d 437 (1966), which held that microfilming of scientific processes with the thief's own equipment and asportation of those copies were proscribed as theft of "goods." The court in agreeing with the government's position, noted that, in *Digilio*, there was no memorization of the information nor copying by the use of the thief's own equipment. One of the criminals actually used government time, equipment, and supplies to make the copies. Finally, the court stressed that a duplicate copy is a record for purposes of the statute and duplicate copies belonging to the government were stolen.

In *United States* v. *Lambert*, 445 F. Supp. 890 (D. Conn. 1978), a Sect. 641 (larceny) case, the defendants were charged with selling information derived from a computer within the Drug Enforcement Administration, Washington, D.C. The information included the identity of informants and the status of government investigations into illegal drug traffic. Only the information, not the documents containing the information, were transferred. The defendants contended that Sect. 641 was applicable only to tangible items, such as documents embodying the information, not the information itself. However, the court held that the open-ended "thing of value" phrase of the statute evidences an intent to cover a wide variety of conduct.

The court saw no reason to restrict the interpretation of Sect. 641 to its common law origins. It held that Sect. 641 should cover larceny as well as any new situations that may arise under changing modern conditions and not envisioned under the common law. The court agreed with the government that the property involved was highly sensitive and confidential information maintained in computer records and had a value only so long as it remained in the government's exclusive possession. It thus held that the phrase "thing of value" in conjunction with the explicit reference to records in Sect. 641 covers the content of such record.

18 U.S.C. Sect. 659 (Theft of Goods or Chartels Moving as, Which Are Part of or Which Constitute Interstate Commerce): Programs may be sent by interstate common carrier. When they are, Sect. 659 protects them from theft, irrespective of ownership. Unlike Sect. 641, Sect. 659 does not seem to proscribe unauthorized copying per se of programs. Although the statute uses "conversion," it is relevant only to the intent of the actor, and not his act, which must be embezzlement, stealing, etc. The most interesting question posed by Sect. 659 concerns theft from interstate commerce.

An excellent discussion of the elements and breadth of what constitute interstate commerce in Sect. 659 is found in *United States* v. *Astolas*, 487 F.2d 275 (2d Cir. 1973). In rejecting appellant-defendants' claim that the trucks they hijacked were not yet, or had ceased to be, part of interstate commerce, Judge Medina quoted with approval the trial court's instruction:

> The interstate character of a shipment commences at the time the property is segregated for interstate commerce and comes into possession of those who are assisting its course in interstate transportation and continues until the property arrives at its destination and is there delivered either by actual unloading or by being placed to be unloaded [487 F.2d at 278].

The requirement of the existence of interstate commerce relates to the time of the theft, *United States* v. *Tyers*, 487 F.2d 828, 830 (2d Cir. 1973), so that one who steals a program may not pass it off later to an accomplice leaving the accomplice immune. Nor is it essential that the program ownership be by common carrier to be protected; it is clear that Sect. 659 covers carriage by the owner [*Winer* v. *United States*, 228 F.2d 944, 947 (6th Cir. 1956), cert. den. 351 U.S. 906 (1956)]. It is equally clear that interstate commerce does end sometime, cf. *O'Kelley* v. *United States*, 116 F.2d 966 (8th Cir. 1941) (theft from boxcar after delivery and partial unloading), but so long as initial steps have been undertaken, cf. *United States* v. *Sherman*, 171 F.2d 619 (2d Cir. 1948) (labeling and delivery of bales of duck canvas to wharf), the program is enroute, cf. *United States* v. *Maddox*, 394 F.2d 297 (4th Cir. 1968) (brief pauses in interstate journey are included within Sect. 659), or yet to be unloaded, Sect. 659 is applicable.

18 U.S.C. Sect. 2314 (Interstate Transportation of Stolen Property): Unlike Sect. 659, Sect. 2314 apparently requires that the stolen property cross state lines. It does not seem sufficient merely for the stolen property to be introduced into interstate commerce. Although there are no reported cases directly on point, that is, where the stolen property was delivered to an interstate carrier but did not actually cross state lines, statutory analysis in *United States* v. *Roselli*, 432 F.2d 879, 891 (9th Cir. 1970), supports this conclusion. In *Roselli*, the court contrasted the antiracketeering statute, 18 U.S.C. Sect. 1952, with Sect. 2314, noting that use of interstate facilities or participating interstate travel was sufficient to provide jurisdiction for the former, while failing to assert that use of interstate facilities was sufficient to trigger the latter. Moreover, reported cases, involving Sect. 2314, have all involved the crossing of statelines. See, for example, *United States* v. *Sheridan*, 329 U.S. 379 (1946) (causing fraudulent check to cross state lines); *United States* v. *Hassel*, 341 F.2d 427 (4th Cir. 1965) (causing victim of confidence game to cross state line); *United States* v. *Jacobs*, 485 F.2d 270 (2nd Cir. 1973) (causing stolen Treasury bills to cross state lines).

The major issue raised by Sect. 2314 is whether a copy of a program stolen, converted, or taken by fraud and transported across state lines can trigger Sect. 2314. The only reported case of a copy used in a related prosecution is *United States* v. *Lester*, 282 F.2d 750 (3d Cir. 1960), cert. den. 364 U.S. 937 (1961). In *Lester*, a coconspirator made numerous copies of valuable geophysical maps, and transported the copies across state lines; the appellant was arrested and convicted for conspiring to transport stolen maps in interstate commerce. Rejecting the appellant's claim that copies were not stolen property, the court held that the property stolen was the valuable idea, not the paper embodiment [282 F.2d at 755].

Although the court in *Lester* found no need to elaborate on its holding, it could have cited *United States* v. *Handler*, 142 F.2d 351 (2d Cir. 1944), cert. den. 323 U.S. 741 (1944), the most thorough analysis to date of stolen property. After analyzing other case law, the meaning of "stealing," and the legislative history of the National Stolen Property Act, now Sect. 2314, the court in *Handler* concluded:

> (1) the stolen property need not be taken larcenously, that is, there are no requirements of asportation, tangibility, etc.; and (2) the statute is applicable to any taken whereby a person dishonestly obtains goods or securities belonging to another with the intent to deprive the owner of the rights and benefits of ownership [142 F.2d at 353]. Since a copy of a program will indeed deprive the rightful owner of the benefits of ownership, a copying should create the stolen property necessary to trigger Sect. 2314.

Note, however, that in *United States* v. *Seidlitz*, No. 76-2027 (4th Cir. 1978), the trial judge dismissed a count based on Sect. 2314 because what crossed state lines was electronic signals, which he concluded were not property. Seidlitz was convicted of wire fraud.

In re Vericker, 446 F.2d 244 (2d Cir. 1971), was a contempt conviction against a defendant who would not testify before the grand jury even after having been granted transactional immunity. The problem was that the defendant was granted transactional immunity as to Sects. 2314 and 2315 only. The immunity, however, was not applicable to the crimes suggested by questioning of the prosecutor. Sections 2314 and 2315 deal with the theft and receipt of stolen goods, wares, merchandise, securities or money, not FBI documents, which the prosecutor had been interested in. Although the court admitted that in some circumstances mere papers may constitute goods, wares, and merchandise, citing *United States* v. *Bottone*, 365 F.2d 389

(2d Cir. 1966) cert. den. 385 U.S. 974, 87 S. Ct. 514, 17 L. Ed. 2d 437 (1966), such papers must be well within the normal meaning of goods, wares, or merchandise, that is, property that is ordinarily the subject of commerce. Thus, geophysical maps or secret manufacturing processes are ordinarily the subject of sale and/or license. However, papers showing that individuals are or may have been engaged in criminal activity or what procedures are used by the FBI in tracking them down are ordinarily not bought or sold in commerce, and, therefore, the government did not show that its questions regarding the theft of FBI documents were related to Sects. 2314 and 2315, and, therefore, could supersede the defendant's invocation of the 5th Amendment privilege.

In *United States* v. *Greenwald*, 479 F.2d 320 (6th Cir. 1973) cert. den. 414 U.S. 854, 94 S. Ct. 154, 38 L. Ed. 2d 104, the Court addressed the issue of whether secret chemical formulae or formulations fall within the statutory language of Sect. 2314 "goods, wares, or merchandise." In *Greenwald*, the number of documents containing the formulations was restricted for purposes of competitive advantage, but one set was given to the defendant, a chemical engineer in the sales department, who appropriated them. The testimony at the trial showed that there was an established market for the chemical formulae and formulation, that is, manufacturers shared formulae by sale or license and treated such as assets similar to machinery or equipment. The court cited *United States* v. *Bottone*, 365 F.2d 389 (2d Cir. 1966) cert. den. 385 U.S. 974, 87 S. Ct. 514, 17 L. Ed. 2d 437 (1966) and *In re Vericker*, 466 F.2d 244 (2d Cir. 1971), to hold that, given an established, viable, although limited market in chemical formulation, the lawful appropriation of original documents containing such formulations fell within the meaning of Sect. 2314 because the formulations were "goods, wares, or merchandise."

United States v. *Drebin*, 557 F.2d 1316 (9th Cir. 1977), was a case in which the defendants contended that motion picture photo plays were intangible and could not be considered "goods, wares, or merchandise" under 18 U.S.C. Sect. 2314. The defendants' arguments consisted of claiming that copyrights were intangible property rights, separate and distinct from property rights in the tangible item from which copies are made, and that a copy cannot be acquired by theft, conversion, or fraud because the copyright owner has no proprietary interest in the duplicate of his work. The court rejected these contentions as illogical and contrary to law and held that the copies are goods or merchandise for the purpose of Sect. 2314. Moreover, the court held that the illicit copying of a copyrighted work is no less an offense than if the original were taken.

Finally, in *United States* v. *Jones*, 414 F. Supp. 964 (D. Maryland 1976), the defendant was charged with transportation in interstate commerce of stolen, converted, or fraudulently obtained securities under 18 U.S.C. Sect. 2314. The defendant claimed that the securities were forgeries and not "securities," noting that Sect. 2314 was not applicable to falsely made, forged, altered, or counterfeited representations of obligations of foreign governments or banks or corporations of foreign governments.

The checks, complete with signatures, were printed by computer as the result of tampering by the employee with the data records stored in the computer. The procedure that the employee used was first to enter an improper vendor code listing, then to enter data regarding the specific checks to be issued to that false vendor, then to forward to key punch the documents and accounts payable slips, and finally to command from the computer the processing of a check run where the computer would automatically print the checks to the false vendor. The issue before the court was whether these checks constituted forgeries and thus the defendant's conduct inapplicable for punishment under 18 U.S.C. Sect. 2314. The court noted that where falsity in the instrument is in the content, rather than the manner of making the instrument, it is not a forgery. In this case the checks were not lies "in writing," but rather the unauthorized issuance thereof. The court held that the mere fact that a computer was used was not relevant because it was simply an inanimate and obedient instrumentality used by the employee similar to a check-writing machine or ballpoint pen and thus was not a forgery.

18 U.S.C. Sect. 661 (Theft within Special Maritime and Territorial Jurisdiction): When programs are stolen in a federal enclave as defined in 18 U.S.C. Sect. 7, a violation of Sect. 661 occurs. As in Sects. 641 and 2314, the question again arises whether unauthorized copying is a violation of the statute.

Although it was assumed for analytical purposes earlier that copying is not within the scope of Sect. 661, abroad reading of the statute may well include it. In *United States* v. *Henry*, 447 F.2d 283 (3d Cir. 1971), the appellant was convicted for stealing a boat within the maritime jurisdiction. On appeal, it was argued that the statute was merely a codification of common law larceny, and since the government failed to offer proof that the appellant intended to permanently deprive the owner of his property, the conviction should be overturned. In rejecting the appellant's claim, the court held that the statute was broader than common law larceny. Drawing on the 2nd Circuit's definition of "to steal" in *Handler*, the court concluded that when one "willfully obtains or retains possession of property belonging to another without the permission or beyond any permission given with the intent to deprive the owner of the benefit of ownership," 447 F.2d at 286, an offense was made out under Sect. 661. As noted earlier, the "deprivation of benefit" theory should enable a prosecutor to support an indictment for unauthorized copying.

Miscellaneous Theft and Theft-Related Offense

Although there is no general federal statute prohibiting theft by false pretenses, except 18 U.S.C. Sect. 1025 (false pretenses within the special maritime and territorial jurisdiction) and Sect. 287 (making false claim to United States), courts have construed Sect. 641 to include false pretenses. See *Burnett* v. *United States*; *Morgan* v. *United States*, 380 F.2d 686 (9th Cir. 1967) (tax fraud as theft of government money by false pretenses). Thus, there seems no bar to charging one who fraudulently obtains computer usage from the United States, while stealing programs, with a violation of Sect. 641.

Many theft statutes, such as Sects. 641, 659, and 2314, have receiving stolen property provisions as well. In addition, Sect. 662 prohibits receiving stolen property within the special maritime and territorial jurisdiction. Section 2315 proscribes the receipt of goods stolen from interstate commerce. Thus, one who induces the theft of programs not only may be charged as a principal, 18 U.S.C. Sect. 2, or as a conspirator, 18 U.S.C. Sect. 371, but also may run afoul of the foregoing sections.

Numerous federal statutes are designed to cover specific types of theft, but they may be applicable to certain instances of program abuse. For instance, if one has the misfortune to steal a program used in the payment of government money, he violates Sect. 285 that deals with taking or using papers relating to claims. If a government employee wrongfully converts, cf. *Morissette* v. *United States*, the property of another that is entrusted to him, he commits an offense under 18 U.S.C. Sect. 654. This section would be particularly effective when the employee provided a copy to an unauthorized third party. Theft of programs from federally insured banks and financial institutions is covered by 18 U.S.C. Sects. 655-657, although there is some doubt as to whether nonmonetary property is covered by Sect. 656 because the protected res is "moneys, funds, or credits," in contrast to "other property of value," 18 U.S.C. Sect. 657. But this loophole is closed by 18 U.S.C. Sect. 2113(b), which covert the theft of "any property . . . any other thing of value . . ." from a bank or savings institution. And finally, if a thief "steals, purloins, or embezzles" property "used" by the Postal Service, he violates Sect. 1707.

Abuse of Federal Channels of Communication

18 U.S.C. Sect. 1341 (Mail Fraud): The mail fraud statute has two essential elements: (1) one must use the mail for the purpose of executing or attempting to execute, (2) a fraud or a scheme to obtain money or property under false pretenses. The courts have been generous in their definition of what is a fraud. The classic statement on this count was made by Judge Holmes, "[T]he law does not define fraud; it needs no definition; it is as old as falsehood and as versatile as human ingenuity." *Weiss* v. *United States*, 122 F.2d 675, 681 (5th Cir. 1941), cert. den. 314 U.S. 687 (1941) (construction scope of fraud in predecessor to Sect. 1341. *Weiss* was quoted with approval in *Blachly* v. *United States*, 380 F.2d 665 (5th Cir. 1967) (referral selling plan as fraud) and *United States* v. *States*, 362 F. Supp. 1293 (E.D. Mo. 1973) (ballot box fraud in primary election as mail fraud), aff'd 488 F.2d 761 (8th Cir. 1973) (see cases cited therein), cert. den. 417 U.S. 909, 417 U.S. 950 (1974).

Thus, the thrust of the various court opinions would include any scheme to copy programs as a scheme to defraud, and any mailing in furtherance of the scheme would trigger the statute. If the thief uses a mailing to defraud a computer center through services, labor, credit, etc., *United States* v. *Owens*, 492 F.2d 1100 (5th Cir. 1974) (mailings which led to receipt of goods on credit as mail fraud), or uses the mailing to obtain the program itself, he falls within the scope of Sect. 1341. The prosecutor should always explore Sect. 1341's applicability in any instance of computer abuse. For a prosecutor's opinion of the effectiveness of Sect. 1341 and, in contrast, the ineffectiveness of the Proposed Code, see Givens, *The Proposed New Federal Criminal Code*, 43 N.Y. St. B.J. 486, 488-494 (1971) et passim.

National Security Offenses

18 U.S.C. Sect. 793 (Gathering, Transmitting, or Losing Defense Information): This section, and those that follow in this category, is of limited use in software abuse. But, as a general rule, whenever abuse involves classified, restricted, or defense programs, these sections should be inspected for applicability. Section 793 is broad in scope; Subsection (a), the geographical intrusion provision, covers property owned, controlled, or used by contractors of the government when the property is related to or connected with national defense. The section also proscribes copying of defense information, unlawful reception, communication of contents, and grossly negligent losses. This statute has been held sufficiently definite to satisfy due process requirements, *Gorin* v. *United States*, 312 U.S. 19 (1941), and has been held to encompass "related activities of national defense" as well as military enclaves [312 U.S. at 28]. See also *United States* v. *Drummond*, 354 F.2d 132, 151 (2d Cir. 1956) (upholding jury charge in same language).

18 U.S.C. Sect. 794 (Gathering or Delivering Defense Information to Aid Foreign Government): This statute provides more severe penalties for actual transmission of the defense information to a foreign government and also includes a conspiracy count. One caveat should be mentioned in this discussion of Sects. 793 or 794, or companion statute Sect. 798, which deals with disclosure of classified information. Although public information has always been outside the scope of the protected res [see *Gorin* v. *United States*; see also *United States* v. *Heine*, 151 F.2d 813 (2d Cir. 1945) (officially disseminated information, no matter how painstakingly culled and digested, is not "defense information")], the Pentagon Papers case, *New York Times Co.* v. *United States*, 403 U.S. 713 (1971), now makes it clear that mere classification is not enough. The flavor of the Black, Douglas, Brennan, and Marshall opinions is that, even in criminal prosecutions, lack of substantial injury to national security might be a valid defense. Even though it is true that White and Stewart contrasted civil injunctive (unpermitted) and criminal (permitted) sanctions, there is language in the Stewart opinion that hints at a need for narrowly construed guidelines on classification. Thus, a clear majority in the case would seem to support the proposition that classified material that had no business being classified, such as information related to Department of Defense lobbying efforts, could not support a prosecution under Chapter 37 of Title 18.

18 U.S.C. Sect. 795 (Photographing and Sketching Defense Installations): In 1950, President Truman declared pursuant to Sect. 795 that all military and commercial defense establishments were to be protected against unauthorized photographing and sketching [Exec. Order 10104, 15 Fed. Reg. 597, 598 (February 1, 1950)]. Since the statute covers "graphical representations" of classified "equipment," it is probable that copying classified programs would fall within this section.

18 U.S.C. Sects. 797, 798, 799, and 952: Section 797 deals with subsequent publication and sale of photographs or sketches of equipment denominated in Sect. 795. Section 798, which deals with codes and cryptographic systems, would be pertinent to any abuse at agencies involved in communications work. Section 799 deals with security violations of NASA regulations, and Sect. 952 deals with disclosure of diplomatic codes.

Trespass and Burglary

Criminal Trespass: There is no general federal statute covering criminal trespass. In fact, the only statute that denominates trespass a crime in Title 18 is Sect. 2152, dealing with trespass on fortifications or harbor-defense areas. Section 2278(a) of Title 42 forbids trespass on installations of the Atomic Energy Commission (ERDA). Neither is particularly applicable to trespass for the purpose of misappropriating programs, unless the situs of the trespass is a fortification, harbor-defense area, or DoD installation.

Burglary: The federal burglary statutes are slightly more comprehensive, but not much. Title 18 provides criminal penalties for burglary of a bank [18 U.S.C. Sect. 2113(a)], post offices [18 U.S.C. Sect. 2115], and interstate carrier facilities [18 U.S.C. Sect. 2117].

-18 U.S.C. Sect. 2113(a) (burglary of a bank). Although some states have denominated copying of trade secrets as larceny, it seems doubtful that entry of a bank to copy programs would make out a federal crime, notwithstanding the language "or any larceny" of Sect. 2113(a). *United States* v. *Rogers*, 289 F.2d 433, 437 (4th Cir. 1961) (the language of the statute refers only to common law larceny). The U.S. Supreme Court has rejected a claim that federal criminal law in this case turns on state law [*Jerome* v. *United States*, 318 101, 106 (1943) (state felonies irrelevant)]. Once beyond those restrictions, however, the statute is effective against the most traditional defenses. Privileged entry is no defense; see *Audett* v. *United States*, 132 F.2d 528, 529 (8th Cir. 1942) (entry may include "walking in [with] a stream of customers through the front door . . . in business hours"), nor is breaking an element of the offense. Although burglary statutes were originally designed to protect occupied spaces from crime, occupancy is irrelevant for purposes of Sect. 2113(a) [*United States* vs. *Poindexter*, 293 F.2d 329 (6th Cir. 1961) cert. den. 368 U.S. 961 (1962)].

-Unlike Sect. 2113(a), 18 U.S.C. Sect. 2115 (burglary of post offices) requires forcible breaking as an element of the offense. The only vague term in the statute is "depredation." While the parameters of the term are hazy, depredation is generally held to mean plundering, robbing, or pillaging. See *Deal* v. *United States*, 274 U.S. 277, 283 (1927) (construing similar language in postal regulations).

Similar to Sect. 2115, 18 U.S.C. Sect. 2117 (burglary of interstate carrier facilities) also requires a breaking. Again, mens rea is intent to commit larceny, which would be common law larceny.

Deceptive Practices

18 U.S.C. Sect. 912 (Obtaining Thing of Value by Impersonating an Officer or Employee of the United States): It may often be the case that one who misappropriates software within a federally protected sphere has falsely represented himself as a government officer or employee in order to gain access to the program. There is no requirement that the "thing of value" be tangible, cf. *United States* v. *Lepowitch*, 318 U.S. 702 (1943) (fraudulent acquisition of information about whereabouts of another), and a copy of the program would certainly seem to fall within the definition. The statute must be read broadly to encompass new concepts of "thing of value" for "it was not possible for Congress in enacting the statute to anticipate all devices and schemes which human knavery might conceive in security benefits . . ." *United States* v. *Ballard*, 118 F. 757 (D. Mo. 1902) (meals and lodging are a thing of value).

18 U.S.C. Sect. 1001: When Sect. 1001, the catch-all that deals with all manner of false representations, is compared with Sect. 912, it becomes apparent that the general rule statute carries a much more severe penalty than the specific statute. In addition, Sect. 1001 requires no fraudulent obtaining of a thing of value; a false, fictitious or fraudulent statement, knowingly and willfully made, is enough to trigger the statute. Whatever one may say about the jurisprudential wisdom of the statute, it seems applicable to almost every instance of computer abuse in the federal sphere. For example, programs may not be divulged to unauthorized persons [5 U.S.C. Sect. 552(b)(4) (trade secrets subsection of Freedom of Information Act)].

Therefore, one who fails to identify himself as unauthorized conceals a material fact, whether or not he represents himself as unauthorized. Is active misrepresentation a less serious crime? Moreover, this section applies to both oral and written misrepresentations. See *United States* v. *Zavala*, 139 F.2d; 830 (2d Cir. 1944) (false oral and written customs declaration). It may even be applicable to electronic signals from a remote terminal that falsely represent the sender as one authorized to protected software.

18 U.S.C. Sects. 1005, 1006 (False Entries in Records of Banks and Credit Institutions): Whenever anyone makes a false entry in a bank or credit institution record, with intent to injure or defraud, he runs afoul of Sect. 1005 or 1006. Although both of the statutes are quite fact-specific, they are comprehensive in their respective areas. Since the purpose of the statutes was to ensure correctness of bank records, *United States* v. *Giles*, 300 U.S. 41, 48 (1937) (teller's failure to file deposit slips is equivalent to the making of a false entry), active or passive omissions or commissions are covered.

Considering the purpose noted above, that is, to ensure correctness of bank records, the breadth with which "bank books" has been interpreted, cf. *Lewis* v. *United States*, 22 F.2d 760 (8th Cir. 1927) (minutes of meetings of board of directors were "bank books"), and the need to protect banks from loss, *Weir* v. *United States*, 92 D.2d 634 (7th Cir. 1937), it seems reasonable that computer records should be within the scope of Sects. 1005 and 1006. Thus, any false entry, obliteration, or alteration of computerized bank records would be a violation of either Sects. 1005 or 1006.

Property Damage

18 U.S.C. Sect. 81 (Arson within Special Maritime and Territorial Jurisdiction): Although arson may be only infrequently used as a tactic in computer abuse, the prosecutor should be aware of the scope of the statute. A key question is whether hardware or programs may be included within the phrase "machinery or building materials or supplies." A case arising from the Wounded Knee occupation indicates that the definition of the phrase may be narrowly construed. In *United States* v. *Banks*, 368 F. Supp. 1245 (D.S.D. 1973), the defendant-appellant was accused and convicted of violating Sect. 81 by burning motor vehicles within a federal enclave. Holding that motor vehicles were not "machinery" within Sect. 81, the court through Judge Nichols, invoked *ejusdem generis* and noted the broad interpretation of "machinery" would endanger the statute as too vague, lacking the "requirement of definiteness . . . that a person of ordinary intelligence must be given fair notice that his contemplated conduct is forbidden . . ." [368 F. Supp. at 248]. Thus, a prosecutor might be advised to style any indictment alleging the burning of hardware or software as, alternatively, an attempt to set fire to a building or structure.

18 U.S.C. Sect. 1361 (Malicious Injury to Government Property): Several cases construing Sect. 1361 demonstrate the liberality with which various courts have accepted indictments charging injury in cases of malicious mischief. Section 1361 was somewhat of a dead letter until interference with the Selective Service began to mushroom in the 1960s. It was resurrected as a catchall to encompass otherwise unindictable offenses. For instance, in *United States* v. *Eberhardt*, 417 F.2d 1009 (4th Cir. 1969), the 4th Circuit considered the famous Baltimore blood-pouring case. Father Philip Berrigan and two others were convicted of violating Sect. 1361 in that they poured blood on Selective Service records. In affirming the convictions, the court utilized the cost of restoring the records as the measure of damages. The appellants did not argue that blood pouring was not "injury" within the meaning of the statute. As a result, the breadth of the case is not clear. At its narrowest, it would mean that any temporary physical obliteration, subsequently restored, is an "injury." While the res in most Selective Service cases was government records at least arguably critical to national defense, other cases construing Sect. 1361 show that neither the injury, nor the res injured need be terribly major. See, for example, *Tillman* v. *United States*, 406 F.2d 930 (5th Cir. 1969) (glassdoor at induction station broken by draft resistors); *Edwards* v. *United States*, 360 F.2d 732 (8th Cir. 1966) (plumbing fixture from vacant home); *Brunette* v. *United States*, 378 F.2d 18 (9th Cir. 1967) (dented fender). Putting all of the cases dealing with Sect. 1361 together with the broadest interpretation of *Eberhardt* may enable a prosecutor to argue successfully that an interference with the use of government software is "injury," and the measure of damage is either the cost of restoration or the cost of development when not restorable.

18 U.S.C. Sect. 1363 (Malicious Injury within the Special Maritime and Territorial Jurisdiction): This section differs from Sect. 81 only in its substitution of malicious mischief for arson.

18 U.S.C. Sect. 2071 (Concealment, Removal, or Mutilation of Public Records): Another statute that was resurrected during the Vietnam-protest era, Sect. 2071 should be effective against misappropriation of computerized government records, especially when a traditional larceny charge cannot be sustained, for example, copying via a remote terminal without subsequent asportation. The bulk of Sect. 2071 cases deal with Selective Service records and documents; see, for example, *United States* v. *Chase*, 309 F. Supp. 420 (N.D. Ill. 1970); *Chase* v. *United States*, 468 F.2d 141 (7th Cir. 1972); *United States* v. *Donner*, 497 F.2d 184 (6th Cir. 1974); *United States* v. *Eberhart*, and thus it would be extending case law to include computerized records as a "document or other thing." Such an extension is rational. The purpose of Sect. 2071 "is to prevent any conduct which deprives the Government of the use of its documents, be it by concealing, destruction, or removal" [*United States* v. *Rosner*, 352 F. Supp. 915, 919 (S.D. N.Y. 1972)]. The res protected by Sect. 2071 is not merely documentary or written records, but any type of public record. Cf. *United States* v. *DeGroat*, 30 F. 764 (E.D. Mich. 1887) (emphasizing the thrust of the statute as toward records, not papers). And under the rationale of *United States* v. *Rosner*, dumping or obliterating a computerized record surely deprives the government of its use as much as a blood-pouring, *United States* v. *Eberhart*, a burning, *United States* v. *Chase*, or a mutilation, *United States* v. *Donner.*

Destruction of Property Affecting National Security: The extreme breadth of what constitutes the protected res in 18 U.S.C. Sect. 2153 (willful injury to war or national defense material during war or national emergency) can be seen in its definition in Sect. 2151. War material includes "all articles, parts or ingredients intended for, adopted to, or suitable for . . . the conduct of war or defense activities." Since the mind has trouble visualizing what in the computer industry would not fall within the definition, it seems clear, so long as scienter is proved, hardware and software within the "defense" orbit are protected. Although the statute applies during war or national emergency, the national emergency declared by President Truman in 1950, Proc. 2912, 15 Fed. Reg. 9029 (December 16, 1950), apparently still exists [*United States* v. *Achtenberg*, 459 F.2d 91 (8th Cir. 1972), cert. den. 409 U.S. 932 (1972)].

The only substantial differences from Sect. 2163 is the applicability of 18 U.S.C. Sect. 2155 (willful injury to national defense material), irrespective of war or national emergency.

Although Sect. 1361 may be construed to reach certain interferences with use, at present there is no provision generally applicable to interference with use or "tampering."

Miscellaneous Provisions

Derivative Crimes and Conspiracy: This section covers federal law applications to derivative crimes and conspiracy.

- –Acts that become criminal only because of the criminal acts of another, derivative crimes, are covered in 18 U.S.C. Sect. 2 dealing with aiding and abetting and Sect. 3 dealing with accessorial liability. As a general rule, any action prior to the crime that induces the criminal act exposes the one who induced to punishment as a principal. Any action subsequent to the crime in the nature of assistance exposes the assistant to a charge of accessory after the fact. Thus, a third part who induces a theft of software, while not indictable by Sect. 641, is indictable under Sect. 2.
- –18 U.S.C. Sect. 371 (conspiracy). Although no general statute makes it a crime to defraud the government, it is a crime for two or more persons to conspire to commit any offense or defraud the United States. This leads to an anomaly—the planning of an act, not criminal in itself, may be a crime. The implications for software abuse are enormous. The broad scope of what it means to "defraud" the United States can be seen in the leading case in this area, *Haas* v. *Henkel*, 216 U.S. 462 (1910). In *Haas*, three persons, one of whom was a statistician with the Department of Agriculture, conspired to falsify official reports concerning cotton crops and to divulge confidential information concerning those crops to unauthorized persons in order that they might speculate in the cotton market. While there was no allegation of pecuniary loss to the government, the Court

rejected a motion to quash the indictment in a habeas corpus proceeding, holding:

[I]t is not essential that such a conspiracy shall contemplate a financial loss or that one shall result. The statute is broad enough in its terms to include any conspiracy for the purpose of impairing, obstructing or defeating the lawful function of any department of Government . . . [I]t must follow that any conspiracy which is calculated to obstruct or impair its efficiency and destroy the value of its operations . . . would be to defraud the United States by depriving it of its lawful right and duty of promulgating or diffusing the information . . ." [216 U.S. at 479-480. Accord, *United States* v. *Johnson*, 383 U.S. 169, 172 (1966) (conspiracy by two congressmen to influence the Justice Department)].

A minor and somewhat redundant conspiracy statute, in the light of the gloss *Haas* puts on Sect. 371, is 18 U.S.C. Sect. 286 dealing with a conspiracy to defraud by payment or allowance of false claims.

18 U.S.C. Sect. 1905 (Disclosure of Confidential Information): This section is potentially applicable to computer abuse in two types of situations: (a) Where a government officer or employee discloses or communicates the contents of programs in government custody but owned by a private person; and (b) same as (a), but where the government owns the programs.

–Obviously, the trade secrets of Sect. 1905, makes the disclosure of "custodial" programs an act illegal unless the disclosure is "authorized by law." For purposes of Sect. 1975, a trade secret is ". . . an unpatented, secret, commercially valuable plan, appliance, formula or process, which is used for the making, preparing, compounding, treating, or processing of articles or materials which are trade commodities." *United States ex. rel. Norwegian Nitrogen Products Co.* v. *United States Tariff Commission*, 51 App. D.C. 366, 6 F.2d 491, 495 (1922), rev'd on other grounds, 274 U.S. 106 (1927). See also *Consumers Union of U.S. Inc.* v. *Veterans Administration*, 301 F. Supp. 796 (S.D. N.Y. 1969) (raw data compiled by government agency not a trade secret of companies providing data). The only law presently requiring wholesale disclosure of information is the Freedom of Information Act, 5 U.S.C. Sect. 552, 871 Stat. 56 (1967); however, it does not apply to disclosure of matters which are trade secrets [5 U.S.C. Sect. 552(b)(4)].

–Disclosure of government computer programs. It appears that if the government develops its own programs, such programs must be divulged on demand unless they are classified, 5 U.S.C. Sect. 552(b)(1), or a trade secret. In reality, agencies have been loath to divulge their staff-prepared programs. See, Comment, Public Access to Government-Held Computer Information, 68 N.W. U.L. Rev. 433, 452 (1973). Whether this reluctance is enough to make them trade secrets is doubtful. See *Shapiro* vs. *S.E.C.*, 399 F. Supp. 467 (D.D.C. 1972) (staff-prepared report on off-board stock trading not "trade secret" within 15 U.S.C. Sect. 552 and not prevented from disclosure by 18 U.S.C. Sect. 1905). Indeed, under the definition in *United States ex. rel. Norwegian Nitrogen Products Co.* v. *United States Tariff Commission*, it seems hard to imagine the government having its own "trade secret," unless it is engaged in a marketing operation. Thus, it seems that any disclosure made pursuant to a 15 U.S.C. Sect. 552 request would exempt the actor from Sect. 1905 liability.

References

[1]Susan H. Nycum and Donn B. Parker, "Prosecutorial Experience with State Computer Crime Laws," Gaston, Snow, and Ely Bartlett, Palo Alto, California, and SRI International, Menlo Park, California (1985).
[2]Koba Associates, *Computer Crime: Legislative Resource Manual*, U.S. Department of Justice, Bureau of Justice Statistics, Washington, D.C. (1980).
[3]Much of the text in this subsection was written by William C. Donnino in the commentary to the law.
[4]Kent W. Coulton, *Computer Crime: Electronic Fund Transfer Systems and Crime*, U.S. Department of Justice, Bureau of Justice Statistics, Washington, D.C. (1982). ■

Computer Accessibility for Disabled Federal Workers

In this report:

This report will help you to:

- Become familiar with the provisions of Public Law 99-506, the "Rehabilitation Act Amendments of 1986," Section 508.
- Learn the variety of alternatives for input, output, and documentation that may be needed in order for disabled workers to use computers effectively.

In 1986, Congress passed Public Law 99-506, the "Rehabilitation Act Amendments of 1986." This law, amending the famous Rehabilitation Act of 1973, contains a small section, titled "Electronic Equipment Accessibility," Section 508, which may have significant impact on the design of computer systems and their accessibility by workers with disabilities. The bill became law when it was signed by former President Reagan on October 21, 1986.

The purpose of this report is to inform concerned computer professionals of Section 508, outline the guidelines and regulations pursuant to the law, describe some of the reaction to the guidelines and regulations, and describe some of the challenges for the future in meeting the computer accessibility needs of users with disabilities.

Section 508 was developed because it was realized that government offices were rapidly changing into electronic offices with microcomputers on every desk. In order for persons with disabilities to keep their jobs or gain new employment in the government, Congress decided it was necessary to make provisions to guarantee accessibility to microcomputers and other electronic office equipment. The driving principle behind Section 508 can be found in Section 504 of the Rehabilitation Act of 1973 which states:

> No otherwise qualified handicapped individual in the United States . . . shall, solely by reason of his handicap, be excluded from the participation in, be denied the benefits of, or be subjected to discrimination under any program or activity receiving Federal financial assistance.

This Datapro report is a reprint of "Computer Accessibility for Federal Workers with Disabilities: It's the Law" by Richard E. Ladner, pp. 952-956, from *Communications of the ACM*, Volume 32, Number 8, August 1989. Copyright © 1989 by the Association for Computing Machinery, Inc. Reprinted with permission.

It should be stated off the top that the scope of Section 508 is not as broad as Section 504. In particular, Section 508 only applies to direct purchases by the federal government and *not* to purchases made by all programs receiving government funding.

Section 508 does not specify what the guidelines should be nor does it delineate a philosophy on which to base the guidelines. A committee established by the National Institute on Disability and Rehabilitation Research (NIDRR) and the General Services Administration (GSA), in consultation with the electronics industry, rehabilitation engineers, and disabled computer professionals, worked for a year developing the philosophy and guidelines which will significantly affect the purchase of electronic office equipment, including computers and software, by the federal government, the largest computer customer in the world.

Initial Guidelines

On November 19, 1987 the Initial Guidelines titled *Access to Information Technology By Users With Disabilities* was published jointly by GSA and NIDRR. The purpose of the initial guidelines is "to set a tone and to provide early guidance to industry on the direction in which the government plans to move in the months and years ahead."[1] The GSA solicited reaction to the initial guidelines from all interested parties.

After considerable response to the initial guidelines, on October 13, 1988, the General Services Administration published regulations pursuant to Section 508 as a Federal Information Resources Management Regulation (FIRMR).[2] About the same time the GSA published FIRMR Bulletin 56 which implements these regulations.[3] Bulletin 56 is basically a revision of the initial guidelines based on careful consideration of the comments received. These and other documents helpful to government buyers, vendors, and users are available in a guide titled "Managing End User Computing for Users with Disabilities"[4] which can be obtained by writing to the address or telephoning the number provided in the footnote.*

*Available from Clearinghouse on Computer Accommodation (COCA), Federal Information Resources Management, General Services Administration, 18th and F Streets NW, Washington, D.C. 20405 (202) 523-1906.

Guidelines and Regulations

The FIRMR regulations inform federal agencies of what they are required to do to comply with Section 508. It is worth stating the general policy under which federal agencies must operate:

Federal agencies shall provide handicapped employees and nonhandicapped employees equivalent access to electronic office equipment to the extent such needs are determined by the agency . . . and the required accessibility can be provided by industry. In providing equivalent access to electronic office equipment, agencies shall consider:

i. Access to and use of the same data bases and application programs by handicapped and nonhandicapped employees;

ii. Utilization of enhancement capabilities for manipulating data (i.e., special peripherals) to attain equivalent end results by handicapped and nonhandicapped employees; and

iii. Access to and use of equivalent communications capabilities by handicapped employees.[2]

The regulations require that agencies of the federal government must assess the needs of its current and potential employees with disabilities and incorporate appropriate functional specifications meeting those needs into solicitations to vendors of electronic office equipment. The assessment of needs of an employee is to be developed by the agency in consultation with the impacted employee to assure that the functional specifications are adequate.

FIRMR Bulletin 56 organizes functional specifications into three broad categories: input, output, and documentation. Under each category a list of specifications is provided to give guidance to agencies and individuals. This list is reproduced here to give an idea of what alternatives are currently available.

1. *Input.*

(a) *Multiple Simultaneous Operation Alternative.* It is common for multiple, simultaneous striking of keys to be required for some functions. An alternative for the same function using sequential rather than simultaneous key strikes may be required.

(b) *Input Redundancy.* Some programs require a fine motor control device such as a mouse. An alternative method such as voice input or key strokes may be required.

(c) *Alternative Input Devices.* In some cases, an alternate input device may be needed to replace or supplement a keyboard or mouse. Alternative input methods may include speech input, eye scan, suck and puff, or headtracking. To support such devices, alternative physical ports or connection capability may be required.

(d) *Key Repeat.* Some computer systems generate automatic repetition while a key is continuously depressed. This feature may need to be turned off or modulated.

(e) *Toggle Key Status Control.* Some computer systems provide a status light for toggle keys. An alternative nonvisual feedback for the status of toggle keys may be required.

(f) *Keyboard Orientation Aids.* Tactile overlays or other tactile indicators may be required to orient a person's hands on the keyboard.

(g) *Keyguards.* A keyguard to help stabilize movements and ensure that the correct keys are depressed may be required.

2. *Output.*

(a) *Auditory Output Capability.* Visual output is the most common form of output of computer systems. Auditory output capable of speech with adjustable volume control and headset jack may be required.

(b) *Information Redundancy.* A visual alternative to auditory output such as warning beeps may be required.

(c) *Monitor Display.* Visual output may need to be enlarged, reproduced verbally, or modified in some way. There are at least three factors to be considered:

i. *Large Print Display.* A means of enlarging a portion of the screen for both graphics and text may be required.

ii. *Access to Visually Displayed Information.* The capability to access the screen may be necessary to provide speech or braille output.

iii. *Color Presentation.* An alternative to color as a way of providing information may be required.

Public Law 99-506—October 21, 1986

An Act
To extend and improve the Rehabilitation Act of 1973.

Section 603. ELECTRONIC EQUIPMENT ACCESSIBILITY.

(a) ELECTRONIC EQUIPMENT ACCESSIBILITY.—Title V of the Act is amended by inserting after Section 507 the following new section:

ELECTRONIC EQUIPMENT ACCESSIBILITY

"Sec. 508. (a) (1) The Secretary, through the National Institute on Disability and Rehabilitation Research and the Administrator of the General Services, in consultation with the electronics industry, shall develop and establish guidelines for electronic equipment accessibility designed to insure that handicapped individuals may use electronic office equipment with or without special peripherals.

"(2) The guidelines established pursuant to paragraph (1) shall be applicable with respect to electronic equipment, whether purchased or leased.

"(3) The initial guidelines shall be established not later than October 1, 1987, and shall be periodically revised as technologies advance or change.

"(b) Beginning after September 30, 1988, the Administrator of General Services shall adopt guidelines for electronic equipment accessibility established under section (a) for Federal procurement of electronic equipment. Each agency shall comply with the guidelines adopted under this section.

"(c) For the purpose of this section the term 'special peripherals' means a seal needs aid that provides access to electronic equipment that is otherwise inaccessible to a handicapped individual."

3. *Documentation.*

(a) Documentation may be required in electronic form to be read by an alternative output device.

The emphasis of the functional specifications is not in providing specific solutions to accessibility problems, but in providing concrete objectives to be met by solutions. This allows for new and creative solutions to providing accessibility.

Reaction to Guidelines and Regulations

There was a considerable response to the initial guidelines and a continuing response to the final guidelines and regulations by the computer industry, manufacturers of special peripherals, rehabilitation engineers, consumers with disabilities, and federal government agencies. Some of this response can be found in "Public Law 99-506, 'Section 508,' Electronic Equipment Accessibility for Disabled Workers."[5]

Industry Reaction

The initial reaction of the computer industry to the guidelines and regulations has been mixed. Some companies are very concerned that their products will not follow the guidelines and that it will cost them dearly to upgrade to meet the guidelines. Some companies are relieved that guidelines are finally coming out so they can go ahead with their plans to make their products accessible to persons with disabilities and be assured that their work in doing so will be adequate. Interestingly, some designers of innovative computer software and hardware products are the most concerned because they feel that the guidelines will limit their creativity, forcing them to compromise their designs to satisfy a tiny minority of potential users.

There are a large number of third-party vendors who manufacture and distribute hardware and software products which make computers more accessible.[6] These vendors are concerned that the guidelines might encourage primary vendors to provide "minimal" solutions which satisfy the guidelines, but do not allow the possibility of using their "superior" products. The guidelines make it clear that the responsibility of the primary vendors is to provide "hooks" into their hardware and operating systems which enable third-party vendors to continue making their accessibility products. A considerable effort has been made by the Industry/ Government Cooperative Initiative on Computer Accessibility to provide concrete guidelines on what hooks are needed to enable access.[7]

There are clearly limits to accessibility depending on the abilities of users. Should office computer systems be made accessible to severely mentally retarded users? Should paint programs be made accessible to totally blind users? The regulation is worded in such a way that industry only has to provide equivalent access if the employee can use it and there is a feasible way to provide it. Thus, industry does not have to provide access to databases for mentally retarded users because they would not be using such software and does not have to provide access to paint programs for blind users because there is no way to do that with current technology.

Reaction of Rehabilitation Engineers

Rehabilitation engineers are the designers and implementors of adaptive devices for persons with disabilities. They have long realized that there is an infinite variety of disabilities requiring a large number of choices of adaptive devices. It is a constant game of catch-up. As technology changes new strategies must be discovered to permit adaptive devices to be added. They have similar concerns to the third-party vendors that the guidelines not be misinterpreted by primary vendors. The primary vendors' responsibility is to make it possible to have a large number of access solutions. This will make the job for the rehabilitation engineer easier.

Should office computer systems be made accessible to severely mentally retarded users? Should paint programs be made accessible to totally blind users?

Consumer Reaction

Consumers with disabilities have long recognized that new technologies can be more or less disabling depending on the technology and the individual. For example, the mouse can be more disabling than a keyboard for someone with limited fine motor control. On the other hand, a mouse-like device controllable with head movement might be less disabling than a keyboard for someone with limited use of his or her hands. Consumers have noticed that the trend in computer systems is to involve more and more of the senses and physical abilities of users in both input and output. There seems to be more graphics and less text, more sound for input and output, and more physical ability with mice, joysticks, and simultaneous keystrokes. The problem is that a particular computer system (hardware or software) may *require* too many senses and abilities making it not adaptable for users with disabilities. In general, consumers seem to be pleased that guidelines are out, helping halt the trend toward nonadaptable computer systems. They also share the concern of the rehabilitation engineers and third-party vendors that the guidelines might encourage minimal solutions by primary vendors at the expense of better third-party solutions.

It is clear that Section 508 only applies to procurements of computer systems for office use by the federal government. Some consumers would like the law strengthened to include computer systems purchased with government funds whether or not the purchaser is a government agency.

The "Technology Related Assistance for Individuals with Disabilities Act of 1988," Public Law 100-407 may have the effect of extending the scope of Section 508 to state governments. This law provides funding to states to develop a response to computer accessibility for citizens with disabilities. The law stipulates that states must comply with Section 508 in order to continue receiving these funds.

Reaction of Government Agencies

A number of government agencies, including Social Security Agency, Department of Education, Department of Health and Human Services, and Internal Revenue Service, have already made multimillion dollar contracts which include Section 508 provisions for computer accessibility for their employees with disabilities. Many other agencies are in the process of preparing solicitations which include accessibility. Within the government there has been a concerted effort to keep agencies informed of the new regulations and to assist them in preparing their solicitations and in finding technical solutions to accessibility problems.

However, there is a natural concern about the cost of providing accessibility, as the cost must be borne by each individual agency. What is the limit that an agency will pay to provide computer accessibility to an employee with a disability? The Section 508 regulations are too new to really ascertain this limit. So far, agencies which have included accessibility have done so by putting it in as a very small percentage of a multimillion dollar solicitation.

Meeting Needs of Disabled Users

The percentage of the population of working age who have some serious disability is currently at least 5 percent and is growing. Public Law 94-142 which mandated that children with disabilities have a right to a full education became law more than 10 years ago. As reported by the Department of Education in 1983-1984 more than 10 percent of all children aged 3-21 were served in special education programs of one kind or another.[8] Thus, young adults with disabilities are more educated than ever before. The Rehabilitation Act of 1973 mandated that these adults should not be discriminated against in hiring or promotion. The triumphs of medical science have resulted in more lives being saved, but in many cases the person saved ends up with some disability. Finally, people living longer are able and want to work well into the traditional retirement years. With proper training and work environment, virtually every person with a disability can be a productive worker. In the U.S. there is an ever growing demand for workers with the ability to use computers and software. As pointed out to the author by Jay Brill, these two trends, growing numbers of workers with disabilities and growing demand for workers with the ability to use computers, compound each other into the consequence:

> *Computer systems must be made accessible to users with disabilities in order to maintain or improve the economic health of the United States.*

Thus, meeting the needs of computer users with disabilities is tantamount to meeting an important need of society as a whole.

Providing access to a computer system for a user with a disability can be an exceedingly tricky problem. Disabilities can be placed into four broad categories: physical impairment, visual impairment, hearing impairment, and cognitive impairment. Within each of these categories, there is a wide variety of specific impairments which present barriers to the effective use of a computer system. For example, visual impairment can range from total blindness to extreme near vision to tunnel vision, to extreme peripheral vision, to color blindness. Some people with visual impairments read Braille while others do not. It is not uncommon for someone with a visual impairment to have a physical, hearing, or cognitive impairment as well. Some impairments inhibit access to standard input devices, while others inhibit access to standard output devices. There are hundreds of solutions and many more potential solutions to this wide variety of computer access problems. What is needed is a relatively small set of "hooks" into computer hardware and operating systems which provide a platform for the development of special purpose hardware and software. To keep the number of hooks to a minimum it would be very helpful if there were a set of *de facto* standards which computer manufacturers adhere to for input and output. For example, textual displays generally have a text buffer which can be accessed so the contents of visual display can be fed into an alternative output device such as a speech synthesizer. Access to the text buffer provides a hook which developers can use to build alternative output devices. With bit-mapped displays it is not enough to have a hook into the display buffer which is just an array of bits. What is needed are hooks into text and pictorial contents of the various windows which are currently on display. This is not easily accomplished.

Making computer systems accessible to users with disabilities is a challenging problem requiring the cooperative efforts of the computer industry, third-party vendors of adaptive devices, rehabilitation engineers, and computer users with disabilities. The development of Section 508, the initial guidelines and final guidelines and regulations demonstrate that cooperation is possible. This law applies only to direct purchases by the federal government, where it is most easily enforced. Much more needs to be done, but the future seems bright for a world where computers will be accessible to all regardless of disability.

Acknowledgments

Thanks to Jay Brill, Susan Brummel, William Roth, Larry Scadden, and Gregg Vanderheiden for many helpful discussions about Section 508.

References

[1]U.S. Department of Education. *Access to Information Technology by Users with Disabilities, Initial Guidelines October, 1987,* National Institute for Disability and Rehabilitative Research and General Services Administration, Information Resources Management Service.
[2]"FIRMR Regulation, 41 CFR Parts 201-1, 201-30, and 201-32." Regulation of Information Resources Management Service, General Services Administration appearing in the *Federal Register 53,* 198 (Oct. 13. 1988), 40066-40068. Available from the Clearinghouse on Computer Accommodation.
[3]"FIRMR Bulletin 56." Bulletin to heads of federal agencies from the General Services Administration, September 30, 1988. Available from the Clearinghouse on Computer Accommodation.
[4]General Services Administration. *Managing End User Computing for Users with Disabilities.* Available from the Clearinghouse on Computer Accommodation.
[5]"Public Law 99-506, 'Section 508,' Electronic Equipment Accessibility for Disabled Workers." Panel at CHI '86, Conference on Human Factors in Computing Systems. Paper in the Conference Proceedings, Special Issue of the SIGCHI Bulletin, 1988, pp. 219-222.
[6]Bandenburg, S.A., and Vanderheiden, G.C., Eds. *The Rehab/ Education Technology Resourcebook Series: Communication, Control, and Computer Access for Disabled and Elderly Individuals, Volumes 1, 2, 3, 1986-1987.* College Hill Press. Available from Trace Center, University of Wisconsin-Madison.
[7]*Considerations in the Design of Computers and Operating Systems to Increase their Accessibility to Persons with Disabilities.* A Working Document of the Design Considerations Task Force of the Industry/Government Cooperative Initiative on Computer Accessibility, May 1988. Available from Trace Center, University of Wisconsin-Madison.
[8]U.S. Department of Education. *Summary of Data on Handicapped Children and Youth,* 1985. ■

Planning a LAN Implementation

In this report:

Datapro Summary

Planning and implementing a LAN requires a crew of experienced hands. The individual performs an essential, but slightly different role. And each person's contribution is critical to the team's success. To design the departmental, enterprisewide or global network, the LAN crew must face the tides of new technology, the whirlpools of politics, the currents of support and training, and the whitewater of the organization itself. These guidelines for implementation and support will help the LAN crew pull ahead to victory, and not run aground.

LAN implementation and support are major issues today in industries, from finance to agribusiness, regardless of the organization's size. Traditional systems management has often been at a loss to cope with the new technological scenario. To harness LAN technology across the board, one must understand LAN technology itself as well as organization and management approaches that emphasize human resource factors. Consideration of one without the other is insufficient. This balanced approach seeks to avoid lopsided perspectives that are not unusual in modern technical management, and are particularly devastating in the LAN field.

Architectural Scope

The scope of LANs has expanded tremendously during the past five years. Their breadth is reflected by the three logical subdivisions which are important in organizing LAN implementation and support: department LANs, enterprisewide LANs, and global networks.

Departmental LANs and the LANs of small- to medium-size companies have much in common. They are the "real" LANs, the basic building blocks for more extensive configurations. Departmental LANs consist mainly of workstations, file servers, and print servers. Occasionally more specialized devices such as communication gateways, mail servers, database servers, and bridges are included. Depending on the size of the LAN, cabling could be simple or sophisticated, temporary or permanent. The LAN is usually formed from a single ring or segment, although multiple segments are sometimes used to cover a spread-out area.

Departmental LANs fall within the domain of the LAN administrator, who is the manager of the LAN. This involves a variety of responsibilities, including planning, design, operational support, and applications support. Administration sometimes includes applications development, security, and coordination of technical resources. At worst, the LAN administrator must call for help; at best, he is a hands-on, mini-data processing manager.

"It has been an ongoing battle to integrate the LAN with a sophisticated workstation environment," says Kevin Sheehan, a LAN administrator of NetWare/Token Ring LANs on Wall Street. "The greatest

This Datapro report is a reprint of "The LAN Crew" by Robert Panza, pp. 40-52, from *LAN Magazine* Vol. 5, No. 9, September 1990. Copyright © 1990 by Miller Freeman Publications, Inc. Reprinted with permission.

challenge is working with MIS to expand application functionality far beyond the departmental LAN toward global LAN connectivity that supports worldwide revenue generating operations."

Departmental LANs are highly visible to users at their workstations. In order to plan and implement a network successfully, the LAN administrator and key users must draw on the expertise of LAN gurus. These gurus might be found in the guise of other highly experienced LAN administrators, in-house or outside consultants, expert personnel of LAN service vendors, or a central LAN service group under the MIS umbrella that functions as an objective internal consulting organization.

Small companies most likely rely on part-time, external consultants who are willing to work for a support fee or on an hourly basis. Departments in large companies require dedicated and sophisticated support that is permanent or immediately available. Permanent consulting staff and retainer relationships are the norm.

Highly expert, reliable support personnel who can (and want to) work in the field are hard to come by. Such personnel provide technical support to many different LAN administrators and/or attend to different departmental LANs on short notice. Such "short-order support" is likely to be an intensive experience with emphasis on problem-solving and troubleshooting.

Large companies occupy a lot of building floor space. Whole buildings or large portions of a single building might contain many organizational components, i.e., departments. Company-wide connectivity can be attained by interconnecting the individual department LAN segments or rings via LAN bridges.

There are economic and management incentives for connecting all data communication devices to the enterprisewide LAN. The objective is to provide shared connectivity facility that must be flexible in the dynamic business environment," says Marc Holstein, an expert in connectivity solution design. "They must allow moves and changes between the business units without impacting the connectivity infrastructure. The bottom line is that enterprisewide connectivity lends itself to transparent integration of business functions on an interdepartmental basis across the entire company."

Some communication devices attach directly to the LAN: Gateways for access to asynchronous, X.25, SNA, and TCP/IP services; remote LAN-to-LAN bridges using uniform protocols, such as NetBIOS, IPX/SPX, or MAC; long and short haul routers; mainframe computer via front-end communication controllers, e.g., IBM 37X5; and mid-range processors (minicomputers) and control units, i.e., IBM 3174s. Most of these devices connect to the backbone ring or to specialized isolation rings bridged to the backbone. Others, such as 3174 control units, are normally attached to departmental rings.

The result dramatically impacts each department's LAN capabilities. After the connectivity structure is in place, the available services must be made known to LAN administrators across the enterprise. Once LAN administrators get the hang of the enterprise LAN, they can innovate. For example, LAN administrators running NetWare may want to exchange data across the enterprise with other LAN administrators who are running IBM/Microsoft LAN operating systems, and vice versa. Or perhaps administrators want to provide users with access to NetWare and IBM/Microsoft servers at the same time.

Enterprisewide LANs often involve multiple hardware types and upper-layer protocols. They can be connected building-to-building with LAN-to-LAN bridges and fiberoptic links. Remote LAN-to-LAN bridges are a good way to facilitate building-to-building connections over a wide area, using dedicated links when economically feasible.

Designing and implementing the enterprisewide LAN requires a broad range of expertise. All network disciplines within the enterprise must be tapped. A central network group within MIS and/or an outside consulting firm usually does the planning and design. The substantial investment certainly mandates careful review before beginning implementation. Some connectivity installation work, such as installing the cable plant, can be contracted to outside firms.

International companies with major locations throughout the world want to link LANs together worldwide. Some LANs are extensive and complex; others are very simple. Connectivity must be built through direct links or through wide area network (WAN) facilities, such as public or private data networks. Linked LANs across the globe mean that the data in a LAN application at one location may be accessed worldwide. WAN expertise dominates this frontier in global data communications. In-house LAN and WAN experts must work together with outside service providers in rowing the connectivity boat toward globally linked LANs.

Planning

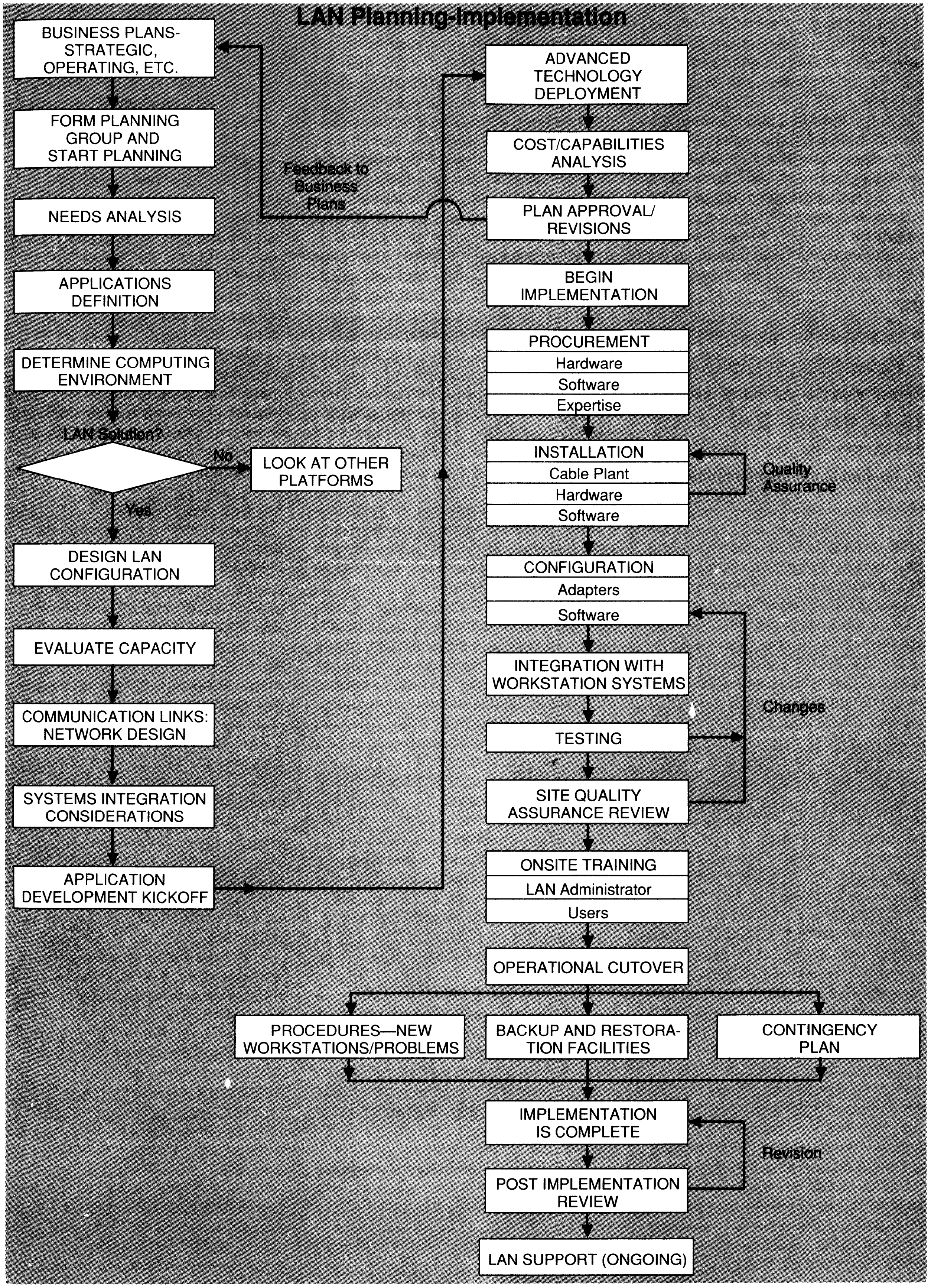
In the real estate business, the emphasis for successful investment is said to be "location, location, location." In LANs, the emphasis is "planning, planning, planning." The planning process is oriented toward design and delivery of a cost-effective business solution, and it follows a pattern largely inherited from traditional systems analysis. When planning is completed, it is always evidenced by a written document or documents. Realism in planning is one of the most important factors required to produce good documentation.

LAN and other systems technology plans must be in concert with the strategic, tactical, operational, and special project plans of the company or business unit. In addition, the schedules for technological deliverables require coordination to ensure that the technology is in place when needed to support realization of the business objectives. The business and technology plans, however, each tend to go their own way. The critical elements of planning for LAN technology are outlined below.

Needs analysis: The business needs must be clearly defined in the language of the businessperson. Sensitivity to the business need driving the requirement vastly increases the probability that the system will serve the users and the business, rather than vice versa. Assessing the business need involves identifying the business problem or problems by interviewing a spectrum of company personnel, objective data gathering and analysis pertinent to the problem, and thoroughly reviewing company operations.

Applications focus: Computers exist for applications. The business need is satisfied through application systems that are supported in a particular computer system environment. Before the system environment can be defined, the analyst must have an idea of which applications will be run, whether outside vendor packages will be used, and/or whether custom systems development will be needed. Applications currently in use might have to be modified, replaced, or migrated to a different system platform. Sometimes new directions in application systems must be

Figure 1.
LAN Planning-Implementation



defined to establish a sufficiently cost-effective computing environment to enhance the competitive posture of the firm.

Computing environment: The application systems are the prime factors that determine the computer environment. The majority of applications run on the pervasive microcomputer-based LANs. While this is the ideal computing environment available today, it cannot exclusively support all application requirements or application systems. Billions of dollars are invested in mainframe and minicomputer hardware, software, and data. Certain specialized applications have been developed to run on only one or a limited number of platforms. Fortunately, it is routinely feasible for all computers to be attached to the enterprisewide LAN. Even if microcomputers are not the platform of choice for a particular application, LAN technology is still likely to be an important part of the solution.

LAN design and configuration: Many questions concerning hardware, software, cabling, topological layout, bridges, communication links, and security must be answered. These take into account all of the elements of LAN planning. Some issues from a seemingly endless list are as follows by departmental, enterprisewide, and globally linked LANs.

Departmental LANs: Decisions include the type of LANs used, network operating systems preferred by the department, the number and type of workstations and servers planned, and the server's speed and capacity in terms of disk storage and concurrent use (e.g., workstation requests per minute). Also, the standard workstation software must be decided, if there will be a standard for each type of user at all. Will the workstation have a Presentation Manager, Windows, New Wave, Saber interface, or a simple menuing system?

Enterprisewide LANs: Issues include the cable type, the logistics of the cabling plant, how rings or segments are bridged, and whether the bridges are controlled centrally or decentrally. The planners must decide which devices will be directly LAN-attached, if there will be mainframe communication front ends (and if so, how many Token Interface Couplers per front-end device), and how many minicomputers, control units, personal computers, and UNIX-based servers and workstations.

Are there interoperability requirements that demand special connectivity, such as Token Ring to Ethernet bridges? Where should isolation rings be placed? Which communication protocols will be used throughout the enterprise? What type of local and remote LAN-to-LAN bridges and data links are required?

Other issues include configuring redundant and backup communication paths, deciding backup and recovery provisions, setting up network administration, evaluating security risks and measures, and potential bottlenecks and traffic loads.

Globally linked LANs: Evaluate the types of traffic and protocols, performance requirements, line speed and quality. Then decide if the economics favor remote LAN-to-LAN MAC layer bridges, long-haul routers, or satellite services. Are dedicated data links justifiable, or must the LANs be linked through public or private packet switching networks? What about ISDN?

Identify the key transport issues. Determine which type(s) of routing should be used and if multiplexers play a role. How must the applications be structured to ensure adequate performance across the global network? Evaluate if it is feasible to deploy advanced technologies, such as database servers and superserver platforms to support the globally shared applications. If so, determine the advantages and if they are sufficient to justify the investment.

Capacity planning: Identify potential bottlenecks that are likely to impact performance sooner or later. Generally, capacity restraints are not due to bandwidth limitations. The most common bottlenecks are server I/O, server file systems, network adapter I/O, gateway capacity, and vendor limitations on the number of users.

Communication links: The WAN experts will consider dial-up, dedicated lines, and X.25 interfaces for interconnection through public or private packet switching networks, ISDN, fractional T-1, E-1, CEPT, T-3, satellite, and microwave.

Systems integration/interoperability: Once the connectivity groundwork is laid, the next objective is interoperability. Interoperability at the network operating system level is fairly straightforward. A more challenging systems integration feat is tying together systems running on Ethernet with systems running on Token Ring to form a logically seamless, distributed computing application.

Application development: If custom applications are likely to be developed, then design consideration must be given to LAN resource functionality, process-to-process communication, application programming interfaces, and other communication-oriented LAN facilities. It is imperative to accelerate the education of application development personnel after finishing the planning process.

Advanced technology deployment: Use of advanced technology should be considered in the strategic technology plan. The capabilities and cost effectiveness of advanced technologies make them very attractive, once they are stable enough for implementation in the operational business environment.

For example, Martin Korsin, a consultant with Logica-Data Architects, at New York Life insurance, plans to reduce cable installation costs "by as much as 80 percent using unshielded twisted pair (UTP) to support IBM 16Mbps Token Ring. Substitution of UTP for IBM Type 1 cabling accounts for the cost differential."

Advanced technologies include 16Mbps Token Ring, high-speed LANs over unshielded twisted-pair, direct LAN attachment of 37X5 controllers, 3174 control units, and mid-range processors, high performance LAN adapters, database servers, the new generation of superservers, and advanced long-haul routers. Soon, implementation of FDDI backbones will be ready for practical consideration.

Cost/Capabilities analysis: My version of cost/benefit analysis is a cost/capabilities analysis. The businessperson wants to know the underlying strategy, the cost, the major deliverables, how the major deliverables quantitatively and qualitatively impact the business, the capacities and capabilities implicit in the technology investment, the risks associated with possible outcomes, and the level of probability associated with each future outcome. It is also valuable to project possible future scenarios against which the plan's strategic validity may be evaluated. Business people appreciate the quantification of business impact over time, taking into account the time value of money. Risk assessment is invaluable since risks abound in technology investment.

Implementation

Upon the plan's completion and approval, we are ready to seize the reins of action. Implementation steps consist of resource procurement, installation, configuration, testing, onsite training, and operational cutover.

Hardware, software, and expert human resources are needed to proceed. Procurement of hardware and software involves the formal ordering process and securing timely delivery. Given the red tape at many large companies, this can be a drawn-out endeavor. At other companies, the plan might have been greeted with enthusiasm, but trauma sets in when it comes time to spend the money. In any event, planners must build sufficient lead times into project plans or find short cuts to ensure that implementation action is not held up due to lack of materials.

A task force of LAN experts can achieve an incredible amount of implementation progress in a relatively short time span. This group should be assembled very early in the game. Once the cable plant is in place, a small LAN can be implemented in as little time as one day; a large LAN takes at least a week or two. The implementation team may be composed of both inhouse and outside personnel. Assemble experts and then give them the freedom to fulfill the implementation objectives. Consultants who are trustworthy, proven veterans should lead the charge. These same consultants should have participated in the planning process since its inception.

Cabling installation is for electricians, not for LAN experts. However, LAN experts can handle the cabling when needed. Little physical layer engineering is normally required at end-user companies because the physical layer is pre-engineered to the vendor's specifications. After the cabling is installed, meters can be used to check for continuity, resistance, and other pertinent electrical characteristics. Electrical grounds may have to be assessed for some network topologies.

Once the electricians have installed the cable plant and checked the readings with their meters, I strongly advise a quality assurance review. The best way to do the quality assurance review is to put wheels on a PC that already has an adapter board installed. Use a boot diskette to initialize network software and then run a few quick network tests at the connection point for each node. Generally, this exercise only takes a few minutes per connection and will most likely uncover numerous problems. The new LAN users will be spared from a negative initial LAN experience.

Installing hardware for direct connection of devices to the LAN is straightforward, although the configuration can be complex. Network adapters must be installed in all PCs along with an adapter cable that connects the adapter to the main cabling plant. TICs and the appropriate cabling would be installed in 37X5s and other SNA devices. For personal computers, the hardware installation is normally a routine PC maintenance job: Put the network adapter board in a slot, attach the adapter cable, run diagnostics, then plug the adapter cable into the cable plant. Hardware-oriented field engineers normally provide the hardware installation services.

Every reputable software package has documentation that gives a clear description of installation procedures. The LAN software installation usually consists of the device driver installation and the network operating system software installation. Device drivers residing in the CONFIG.SYS file (of OS/2 and DOS nodes) enable the higher-level LAN software to access the LAN adapter. Adapter device drivers sometimes come with the adapter or may be part of a separate package.

The IBM LAN Support Program is an example of device drivers in a separate package; those device drivers provide data link control and NetBIOS interfaces for IBM DOS-based LAN operating systems on Token Ring and Ethernet. Normally, the device drivers are just copied onto the node and then an installation aid program is run. The higher-level network operating system software also normally installs through an installation aid program that comes with the package. The exceptions are the NetWare shell files for DOS-based workstations; they are usually just copied onto the workstation. Many installation aid programs automatically update the CONFIG.SYS and AUTOEXEC.BAT files of the LAN node that they are run on. (Automatic updates may or may not be helpful.)

Existing workstation and/or menuing systems may need to be changed in order to take advantage of the LAN functionality. Therefore, new workstation software might be part of the installation task list. A workstation or menuing system should be used, since LAN features such as virtual drives confuse the user at the operating system prompt.

Installation of bridge software is not extremely difficult, but the procedure varies by vendor. For example, an IBM Token Ring bridge attaches to and connects two rings. A NetWare 286 server with internal bridging, which can have up to four different LAN adapters installed, also routes traffic among the dissimilar LANs.

Gateways between LANs and external communication systems must have multiple connections, which means multiple communication adapters. Device drivers and communication software need to be installed on the gateway node. Again, the installation procedure depends on the gateway type and vendor. Although the LAN side is relatively straightforward, consult a WAN expert for the external services side. Routers are similar to bridges and gateways from the installation perspective, but are more powerful and complex. They can potentially support many different kinds of LAN-to-LAN and LAN-to-WAN connections.

On enterprisewide LANs, specialized nodes can be installed to perform functions such as network management, monitoring, capacity utilization, and problem diagnosis.

After the cable plant is installed, little or no configuration work is required. However, adapter configuration settings might require adjustment for a variety of reasons, including performance optimization, resolving conflicts with other adapters, and avoiding memory conflicts with EMS device drivers.

Adapter configuration settings may include any of the following: the range of PC memory addresses and amount of PC memory used by the adapter, hardware interrupt level, primary or secondary adapter designation, data rate, base I/O address, direct memory addressing, and connector type. Settings for a particular adapter vary by type.

Some network device drivers, like those of the IBM LAN Support Program, can be used to set the range of PC memory addresses used by the adapter to establish a locally administered network address for the node, and to configure on-board adapter RAM. For Token Ring, data link transmit and receive buffers, messaging buffers, timing parameters, and resources for supporting multiple node-to-node communication links are the most important factors. Network performance and functionality are impacted by the parameter settings used to configure the adapter.

Configuring network operating system software can be very elaborate for IBM/Microsoft LAN operating systems. For DOS workstations, lots of NET START parameters and heuristics can be set to tailor performance, expand functionality, or reduce the amount of PC memory used by the LAN software. The latest software versions store the parameters in a DOSLAN.INI or LANMAN.INI file. Parameter settings at this higher level must be orchestrated with the parameters coded after the NetBIOS device driver. Tuning the NetBIOS parameters is both an art and a science that few have mastered.

OS/2 workstations and servers running IBM/Microsoft LAN software are two or three times more complex to configure than their DOS predecessors. The IBM OS/2 LAN Requester and LAN Server parameters are stored in an IBMLAN.INI file. In addition, they require the Communications (Comms) Manager to be configured, since all the interfaces that support the LAN Requester and LAN Server are contained in the Comms Manager. In terms of complexity, the Comms Manager configuration is equal to or greater than that of the IBMLAN.INI file.

IBM workstation and server software can be installed and then configured afterward, although some basic configuring is required at installation time. The opposite sequence pertains to NetWare server software. It must be configured before installation; only the maintenance may be done later. The configuration is part of the network generation process. Likewise, NetWare workstation shell files for DOS nodes are configured during the shell generation and installed later. The NetWare Requester for OS/2 installs more like the IBM OS/2 LAN Requester than its NetWare brethren; however, it is quite a bit simpler to configure than the IBM OS/2 LAN Requester.

Workstation systems, menuing systems, and environment software, like Microsoft Windows, impact and are impacted by the LAN software configuration. Other LAN and nonLAN communication software are additional variables pertinent to configuration. RAM requirements and memory addresses might have to be juggled to avoid conflicts and ensure that sufficient RAM is free to run all workstation applications. Adapter resources over and above those required by the LAN operating system might have to be allocated for other communication software that concurrently uses the LAN adapter.

Throughout the LAN configuration process, the nodes must be thoroughly tested first by the task force persons doing the configuration work, and next by the LAN administrator and users. Every possible workstation option should be checked.

The initial network administration is part of the configuration process. Users must be setup, shared LAN resources (file sets, printers, communication devices, etc.) must be defined, access rights to the shared resources must be created, and shared application systems must be installed on the file server. On-site LAN orientation and training should be given to the LAN users as well as to the LAN administrator. The LAN administrator hopeful witnessed and participated in the hands-on implementation process; much of that experience serves as a training exercise.

Documentation of the LAN configuration must be prepared and walked through for accuracy. It is normally very detailed for the cable plant of enterprisewide LANs. Important procedures covered in the documentation for the LAN administrator include implementing a new LAN workstation, backup and restoration of workstations and file servers using LAN-based backup devices, problem solving and trouble shooting procedures, and a contingency plan in the event of LAN failure from a variety of causes.

Once the user group is satisfied that the LAN works, the implementation task force turns the LAN over to the LAN administrator. From this point forward, the LAN administrator relies on the LAN support group whenever help is needed.

Support

All implemented LAN resources must be supported. Broadly speaking, the LAN support organization's mission is twofold: problem solving/troubleshooting and training.

Hardware and software is normally serviced through different groups within LAN support. Hardware support covers the cable plant, network adapters, and physical maintenance of PCs, 3174 control units, 37X5 communication controllers, and other devices that are directly LAN-attached. Routine non-cable plant hardware support should be a subcomponent of the central hardware maintenance group, because the same skills apply to both LAN and non-LAN hardware. Eventually, almost all hardware within the organization will somehow be related to LANs.

The bread and butter of LAN support is software, operating systems, and workstations. More than 80% of LAN support calls pertain to one or more of these. Application system concerns sometimes come into play. Special configuration considerations pertain to applications running on LANs. LAN experts can perform astounding "tricks" with single and multiuser applications on LANs. Also, application developers who are writing LAN applications periodically need advice from the LAN experts on performance optimization, multiuser LAN programming techniques, and programming interfaces.

LAN support can be performed over the telephone, via E-mail, through remote LAN management tools, or on-site. Remote access software helps with support from a distance by allowing central support personnel to view and manipulate nodes on remote LANs or on LANs elsewhere in the building. However, LAN support personnel must appear on-site when the problem cannot be resolved by advising the LAN administrator or by using the network management and diagnostic tools. While the LAN administrator is usually the point of contact for the LAN support group, high priority users sometimes get special attention.

Enterprisewide LAN support sometimes involves multiple network disciplines; global network support frequently involves multiple network disciplines. Multiple network disciplines mean trouble because people in different technical components of the organization must work together. And the different mindsets sometimes seem incompatible. Eccentricity in technical experts is not unusual. This is a key management issue. It is best to structure the support organization in a manner that minimizes interdisciplinary conflicts. Otherwise, problem resolution is hampered, and sooner or later systems management is embarrassed in front of the business operating units.

The LAN support group must always be prepared for contingency action, since it is the support group of last resort.

Training

LAN training requirements are much more extensive than most people realize. They are also ongoing. LAN training is vital to the efficient and successful dissemination of LAN technology throughout the company. Much of the training should be done on a formal hands-on basis in a training facility equipped with a full set of communication technology. The trainers can be drawn from outside LAN experts, and/or the in-house LAN implementation task force and LAN support group. For a large company, an offering of training courses by audience might look like the following.

Upper management: One day seminars for upper management include: Business Planning for LAN Technology, Strategic Deployment of Enterprisewide LAN Connectivity, Why LANs? Economic and Qualitative Advantages of LAN-based Systems, Solutions for Integration of Heterogeneous System Environments, Alternatives in LAN Technology, and Organizing Technical Resources for Effective LAN Implementation and Support. Since top managers are the decision makers and later are the "nurturers" of the LAN-based strategy, many of the seminars should be attended before the planning process begins. Afterward, a one-day hands-on lab can be tailored to the interests of select members of the upper management group.

Middle management: One day seminars are best for middle managers, such as: Departmental LANs—Management and Functionality, Managing the LAN Administrator, and Alternatives in LAN Technology.

LAN administrators: At least two courses are needed: Basic LAN Administration and Advanced LAN Administration. The former is a three-day course; the latter is a five-day course. The courses, which split time evenly between lectures and laboratory workshops, are geared toward the type of hardware, LAN software, operating systems, and workstations implemented on the LANs for which the LAN administrators are responsible.

Users: Two levels are needed. The introduction to LANs lecture lasts for a half day. Advanced issues in LANs is a two-day course, with time divided equally between classroom and hands-on lab.

LAN implementation/support personnel: Basic LAN implementation and Support and Advanced LAN Implementation and Support are both one-week courses emphasizing lab work. Lectures are given in the lab.

Workstation development personnel: The LAN Technology Considerations in Workstation Design course enables workstation development personnel to feel comfortable with LAN technology and understand how to incorporate LAN technology into their workstation products. It is a three-day course that emphasizes lab work.

Application development personnel: LAN Application Systems Development and Design enables application developers to learn LAN programming interfaces, become comfortable with LAN technology, and understand the subtleties of LAN application design. This course mainly consists of three days of lab work.

Physical layer specialist personnel: Personnel who implement and maintain the enterprisewide cable plant and network adapters often get locked into a narrow "physical" view of LANs. A broader perspective helps them in testing and diagnosing physical layer problems, using tools that run in the higher layers, as well as vendor problem determination procedures. Three days of lectures and labs will do the job for this group, although they will probably want more.

Organization

The organization's structure should consciously encourage teamwork throughout the company by maximizing internal efficiencies and minimizing internal and external political wastefulness. It should also honestly involve the business operating units in all major technology decisions, and maintain genuine sensitivity to the viewpoints of non-networking system groups who might need to incorporate network functionality into their deliverables.

Higher level management must root out excessive politics that impair the realization of satisfactory efficiency. Interdepartmental cooperation is imperative. Otherwise, the systems organization will be viewed very negatively from the vantage point of the business operating units—the kiss of death for MIS.

Strong management does not mean a top-down approach that dominates and always sets direction. With fast paced technologies, the top-down approach can be counterproductive and wasteful. Control must be relinquished where a bottom-up approach makes more sense, as is generally the case with LANs. Upper management then assumes the role of nurturers and benefactors. Control can be satisfactorily maintained through their trusted agents. It does not take long for upper management to see the beneficial results of LAN innovation when the game plan is sound and the implementation runs smoothly.

Today there is a general crisis in technical management at American businesses. Being people- or business-oriented is not enough; systems management must understand and must be able to finesse the technology analogous to the way accounting managers understand and can finesse the numbers in financial analysis. Good technical leadership has its roots in technical managers who understand the technology and what is going on around them. Keeping up with technological change is not easy; it requires technical renewal training, involvement, teamwork, person-to-person communication, and most important, using the technology everyday at the office to do one's job.

Good management brings out the best in people. A good manager is a catalyst for teamwork and achievement through which the department as a whole becomes significantly greater than the sum of its individual human elements. People, however, are more important than the organization. An organization is a logical combination of human resources; it is not a higher form of life.

According to Wall Street legend, J. Pierpont Morgan valued human resources above all other assets—buildings, bonds, stocks, real estate, technology. Morgan understood that human spirit is the ultimate source of all achievement. The organization that harnesses human spirit has a vast advantage over those that ignore or debase it.

For Team Players

LAN implementation and support is a field-oriented mission. Coming to grips with LAN implementation and support in a large organization starts at the top—with the understanding that LANs are a bottom-up technology. Reorganization of more traditional systems departments is most likely necessary. On the organization chart, control must be relinquished downward. Technical experts can get the ball rolling, but that alone is not satisfactory for disseminating LAN technology throughout the company. Training is a good tool to spearhead dissemination, but it must target numerous groups and different levels across the business.

Keeping up with the fast-paced network technology is another matter. A methodology for doing so can be worked out for each individual firm. Once the methodology becomes part of management practice, it must be periodically revised (at least once a year) for adaptation to changes that have taken place in LAN technology and in the business. ■

The Four Phases of LAN Implementation

In this report:

Datapro Summary

LAN planning, implementation, and maintenance is a complex, lengthy process that often evolves with twists and unfortunate surprises. To set forth an efficient and cost-effective LAN system, managers should follow a structured series of steps. The author of this report outlines the steps as: Study, Selection and Design, Implementation, and Operation.

The Study Phase

Companies, corporations, and organizations are rapidly recognizing that information is a key resource. Today's corporations are relying more and more on computer-based information systems to manage this burgeoning data resource. However, computer-based information systems cover a wide range of hardware and software. They can be as small as a single personal computer or as large as multiple mainframe computers. The fastest growing computer-based information system being implemented to meet today's business needs is the local area network.

To be effective, the local area network should be implemented according to a planned, systematic series of steps and procedures. This ensures an efficient and proper installation. These steps and procedures are divided into four broad basic categories or phases (see Figure 1), which taken together form what is known as the life cycle or system development cycle of a computer-based information system:

1. Study
2. Selection and Design
3. Implementation
4. Operation

This section discusses the study phase of the system development cycle. Subsequent sections discuss the remaining three phases.

The study phase is initiated to ascertain the need and/or justification to implement a local area network as the computer-based information system for an organization, company, or corporation. Essentially, the study phase consists of two separate sub-phases (see Figure 2):

- Investigation and analysis
- Feasibility study

After these two sub-phases have been completed, the project may enter into the next major phase of the LAN life cycle.

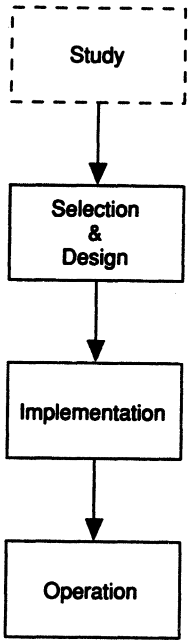
Investigation and Analysis

The investigation and analysis portion of the study phase consists of information gathering, followed by a documented report. The necessary facts and information can be obtained by following the procedures outlined below:

1. Collection of background information
2. Definition of problems
3. Assessment of information or user requirements

This Datapro report is a reprint of Chapters 10 through 13, "Implementing the LAN: The Study Phase," "Implementing the LAN: The Selection and Design Phase," "Implementing the LAN: The Implementation Phase," and "Implementing the LAN: The Operation Phase," pp. 145-186, from *LAN Primer* by Greg Nunemacher, Copyright © 1990 by M&T Publishing, Inc. Reprinted with permission.

Figure 1.
System Development Life Cycle



4. Identification of resources and constraints
5. Report of information collected

Step 1. Collecting Background Information

This is the first function of the investigation and analysis process. Information must first be gathered about the organization before any true analysis can be performed. This information should include the organization's history, its present status and projected growth, operating policies, management styles, office environment, unique components of the organization, an overview of present office systems and procedures, and the attitudes or viewpoints of the people who will be affected by the local area network. This last element, assessing the views of the users, is very important and should not be overlooked or taken too lightly. The cooperation or non-cooperation of the people involved can "make" or "break" a system.

A good understanding of the company's politics is also very useful. This gives the analyst a better understanding of what he or she may encounter in trying to implement a local area network. For example, a key executive in the company may have come from another organization that used a minicomputer exclusively. This person could be totally opposed to the installation of personal computers throughout the company. At the same time, other employees may like using their personal computers. To these people the idea of being connected to a local area network may be equated to taking away some of their independence since network information is centralized and with the LAN these people will have to depend on others to support their daily operations. As far as company politics is concerned the local area network may be the brainchild of one manager, and certain people may want to see that particular manager in as much turmoil as possible. In this situation, cooperation may not be forthcoming for the LAN installation, not because the local area network is ineffective, but simply to cause problems for one unpopular manager.

Step 2. Defining the Problem

As discussed later, initiating a study to examine the feasibility of installing a local area network is not done on a whim. There usually exists concrete problems in the organization's information processes of sufficient magnitude to warrant systems analysis. Many times, these problems do not show themselves openly; however, they must be identified so that the faults in the information processing system are corrected. Using the background information that has been collected, it may be easier to identify and define these problems.

Step 3. Assessing Requirements

Any computer-based information system that does not supply timely and accurate information is of little use. What's more, the information that is produced must be of value to the people who need it. Twenty reports may be offered to one small department, but if these reports do not provide correct information, what good are they?

Painstaking care must be taken to ensure that the information requirements of the organization and its workers are met. This requires detailed analysis. The volume of information processed by the computer-based information system must be known. What departments need what type of application programs? What departments need what reports and in what formats? The tasks that should be executed on the LAN and those that should be handled locally on the personal computer must be assessed. Finally, the expansion of the organization must be considered—what information needs will exist in three months, six months, one year, two years, five years? These growth needs must be projected and documented.

The overall, long-term requirements of the people and the organization must be met in order to make a local area network a viable solution. Without meeting the requirements of the people and the organization, the most sophisticated LAN is useless.

Step 4. Identifying Resources and Constraints

During the investigation and analysis phase, the organizational resources must be identified and documented. Conversely, the organizational constraints and limitations must also be documented.

Organizational resources that will have an impact on implementing a new computer system essentially fall into two categories:

- Computer hardware and software resources
- Human resources

The first thing to examine are the computer hardware and software the organization already possesses. Does each person have his or her own personal computer? Does the organization have a minicomputer or a mainframe computer? If so, are all computer users connected? If they are connected are they connected by PC terminal emulation or are they using dumb terminals? The software applications that the organization presently possesses also must be inventoried. Full knowledge of what the organization does and does not have is very important, and makes it possible to make a more accurate assessment of the organization's true needs. It will also help to accurately estimate costs.

The persons affected by implementing a local area network also must be identified. Their level of skill and their attitudes toward computers and computer applications should be known. This is necessary to determine the amount of training required, to identify who will support the local area network, to decide whether additional programmers will be needed, and for "public relations" purposes to ensure a smooth implementation.

Organizational constraints and limitations can take many forms, including financial resources which are too often the primary constraint. During the investigation phase, a budget for the local area network must be established. Another major constraint is the lack of personnel available to maintain and manage the local area network. When other constraints are encountered, such as unsuitable conditions to run cable, they must be documented.

Step 5. Writing the Report

After these initial investigations are completed, a quantity of information and facts will have been compiled. This information must now be organized and thoroughly documented. This documentation can then be used for the next phase of study, the feasibility study.

Feasibility Study

All the information that has been collected from investigation and analysis can now be applied to ascertain the practicality or need, in other words the feasibility of installing a local area network. Will the local area network fulfill the requirements identified by the investigation and analysis? The feasibility study should answer that question.

Cost and Benefit

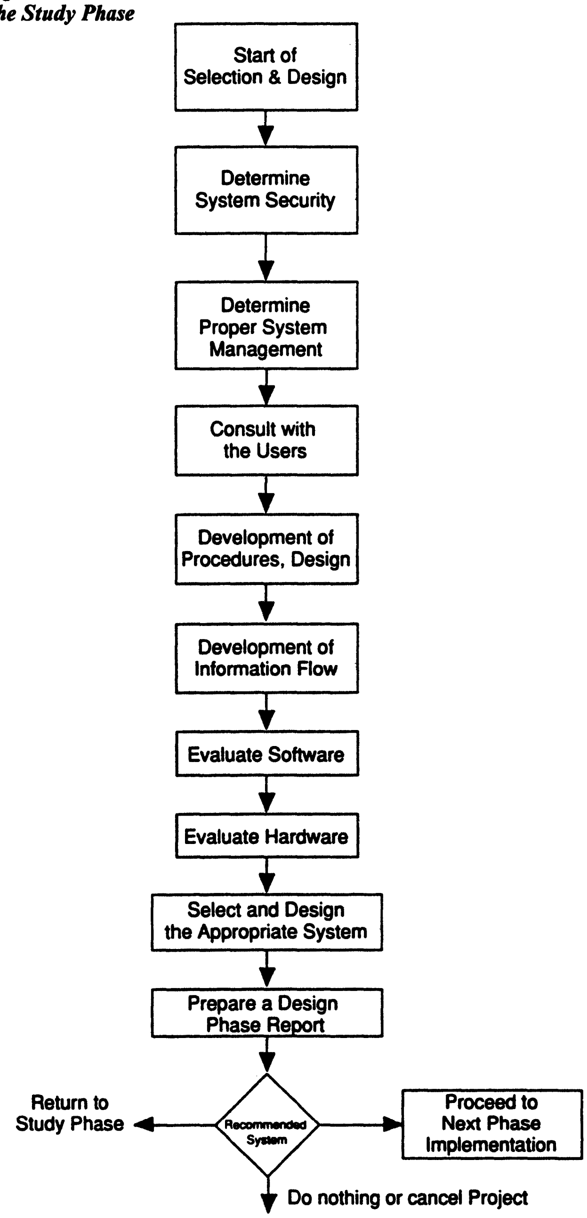
The feasibility study must estimate the actual costs of implementing a local area network. It should also show, in dollars, the benefits gained from the local area network to justify the costs. It might be useful to use time tables showing how costs are recouped, including amortization schedules.

Study Phase Report

The feasibility report is drafted next using the information gathered during the investigative process. This report should define all associated details that were uncovered during the investigation and the results of the feasibility study. All of this accumulated information is studied and applied to assess the capabilities needed and possibilities offered by a local area network. If any information specific to a local area network has not been gathered during this feasibility study, the analyst must then obtain this additional information. Lastly the report should offer a recommendation.

Of course, there may be a host of unanticipated factors and issues that must be addressed when conducting the study to determine the feasibility of installing a LAN. There may be more investigation and analysis than was originally thought. Not only must the needs and problems of the organization be identified, but the analyst must know how LANs operate and if a LAN can be of benefit. More than one analyst may be involved in this first phase. One analyst may determine or identify all the problems, existing procedures, constraints, limitations, and resources. Then a second analyst who is very conversant in LAN technology and applications may be recruited to write the recommendation using the information obtained by the first analyst. One must know what a LAN can and can not do before one can write any sort of recommendation. This is very important for any successful LAN feasibility study, or any other subsequent phase of the LAN development. Without sufficient knowledge and experience, it is impossible to make authoritative recommendations and guide others in nurturing the LAN through its life cycle.

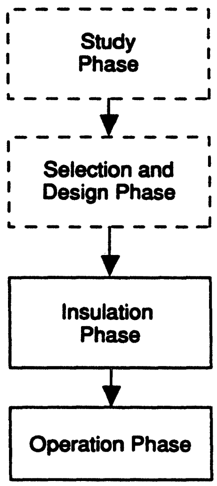
Figure 2.
The Study Phase



As always, he careful who performs the feasibility study. If a vendor performs this study, they can slant or bias the information so the recommendation appears favorable for a specific local area network. It is important to know at all times who is performing what portion of the study, and what vested interest that person might have in the study's findings.

The study phase report is usually presented in a meeting so those connected with the LAN project can review the report and discuss its findings. Three actions may result from this review meeting: One, the project is postponed or rejected. If this is the case, all investigation and analysis stops. Two, it is decided that more information is needed before a final decision can be made. This results in more investigation and analysis. Three, the review committee endorses the recommendation and the life cycle of the network then proceeds to the next phase.

Figure 3.
Selection and Design Phase



To Install or Not to Install?

In most cases, there are specific reasons that trigger an organization to investigate the feasibility of installing a LAN. It is unusual for a LAN to be implemented for no significant reason (although this has happened—to be "in vogue," someone decides the company needs a LAN whether it is justified or not). Attention should be paid to underlying reasons and causes that prompted the initial investigation and analysis. The entire information processing structure should be brought down to the very basic level. First, identify existing problems and needs that are not being met, and then identify future needs that need to be addressed as the company grows. Secondly, consider if the LAN fulfills those needs in the present, and will it be able to grow to accommodate future needs.

Using this basic premise, there are some easily identifiable reasons to initiate a feasibility study, including:

- Improved efficiency
- Improved control
- Improved productivity
- Improved services
- Costs savings

By understanding why the system study was initiated, the analyst will be better able to define the prime directive driving the investigation and analysis. Knowing the prime directive behind the system study makes it easier to focus on items that might be otherwise overlooked, and which would probably affect the feasibility study by their absence. Should some items be overlooked and slip by in the feasibility study, additional problems may arise as the system is developed. Oversights could also cause the project to be canceled when it should not be canceled.

Improved Efficiency

Creating a more efficient operation could be a major reason to start a system study. The analyst must look at almost every aspect of the organization to be able to improve its efficiency, which requires substantial investigation and analysis. All systems currently in place, all procedures presently used, all plans for expansion must be examined. One particular catalyst that creates inefficiency and substantiates the need for a local area network is the growth of the organization. What once started as a small endeavor can in a short time grow into a sizable enterprise. Implementing a LAN may be one way to use outmoded equipment. The organization probably has some microcomputers and peripherals that could be incorporated into a network.

Previously, in larger organizations, when a department expanded, moving to a centralized mainframe appeared to be the only viable solution. Unfortunately, a centralized mainframe could not always provide the end users with timely and precise information. Today, local area networks are considered an alternative solution.

Improved Control

Increasingly, there is a trend among organizations for individual departments to take on the information processing control for their groups. The managers prefer this to having everything stored on one, centralized, computer-based information system. Localizing information processing capabilities gives department managers more security and more control over their department.

Where a departmental minicomputer was the alternative to a centralized system, the development of sophisticated personal computers with more processing power has taken the place of minicomputers. This is especially true when the data is provided via a local area network. So today's departments are installing local area networks instead of minicomputers.

Improved Productivity

A primary objective of any manager is to increase the productivity of his or her staff. One tool that has proven its ability to improve efficiency time and again is the computer. However, even in today's business environment, information available from a centralized computer cannot always provide data when it is needed and in the form it is needed. So, again, whether the work group is large or small, managers are turning to departmental processing. The local area network is one way to assist in the increase of departmental productivity.

Cost Savings

A local area network may be warranted on the basis of cost savings alone. Usually it is less expensive to implement a local area network than to install a minicomputer or small mainframe.

Microcomputers have greatly reduced the cost of information processing. No longer do small companies have to rely on expensive MIS hardware or outside data processing companies. They can now do the same data processing inhouse using microcomputers. When the conditions are correct the company can use their existing microcomputers to form a local area network that increases their data processing power. Thus the existing equipment is capable of being "recycled" into a local area network.

Improved Service

Many companies, and even departments within large organizations, rely on their customer database for their existence. In service businesses, for example, to be competitive a company has to increase the type and quality of service, and computerized data processing is often the key.

Local area networks can provide the information processing requirements that a company needs to improve their service to their customers. For example, in a busy doctors' office, information on different patients may be accessed from different terminals by different nurses, which decreases the time the patient must wait and thereby provides a better service.

Local area networks are not the only solution available to solve information processing problems. They do, however, offer a timely and appropriate solution in many, if not most, cases. Whether a local area network is the appropriate solution for your environment depends on what you uncover during the study phase of the system life cycle.

The Selection and Design Phase

Beginning the LAN design phase implies that the recommendations put forward in the feasibility study were approved (see Figure 3); the benefits gained from implementing a local area network will substantiate the costs.

The design phase begins the process of creating a detailed blueprint of the system. Much of the information collected during the study phase also can be used in the design phase, but a greater degree of detail will be required. First and foremost, the system must be designed so that it falls within budgetary constraints. The system must also satisfy those requirements identified in the study phase.

Listed below are some of the important issues that must be addressed during the design phase (see Figure 4):

- Consult with the users.
- Develop a chart showing information flow.
- Select the best topology.
- Select the appropriate transmission media.
- Evaluate the available software.
- Evaluate the available hardware.
- Determine the degree of system security required.
- Determine the proper system management.
- Select and design the most appropriate system.
- Prepare a design phase report.
- Develop a procedural design.

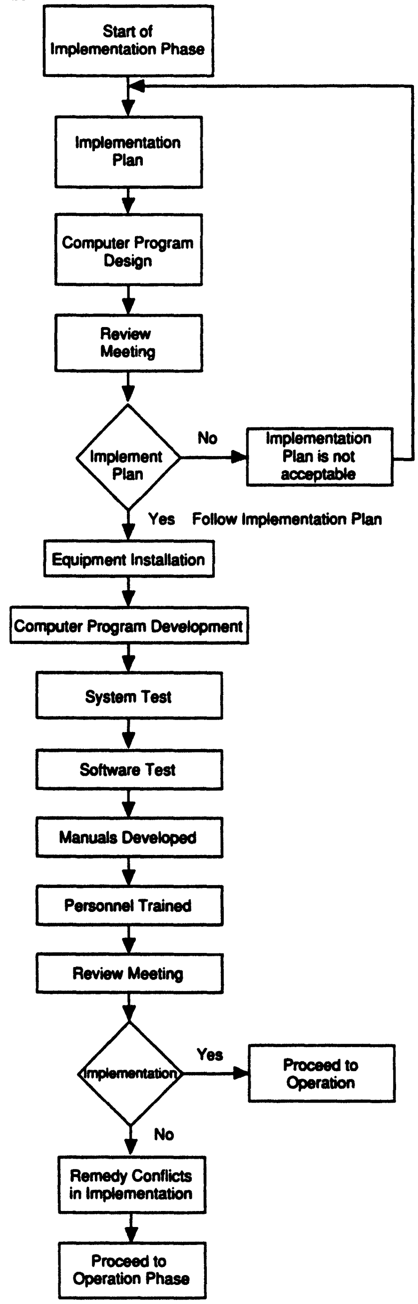
System Security

Many times, when a computer-based information system is developed that consists solely of PCs, the security aspect is not taken seriously. The biased viewpoint remains that PCs are not serious data processing equipment and therefore security is often overlooked.

A PC may contain data that is just as confidential as data found on a mainframe computer. Once the PCs are interconnected, the issue of security looms even larger, and local area network security is something that should be taken seriously. The choice of system components and the system design should reflect the degree of security needed. After all, an organization's fate may be determined by information security. Harm may come to the organization whose information is not secure and may be accessible to those who should not have access. System availability must be monitored carefully.

A local area network's security needs are much like the security needs on a minicomputer or a mainframe computer. The LAN, like the mainframe and minicomputer, may store confidential data. This data needs protection regardless of the system used. Like the mainframe and minicomputers, a LAN stores many application programs and data that are not for everyone's use. The degree of accessibility given to each user must be determined.

Figure 4.
Design Phase



Another security issue that is unique to a local area network is the accessibility of the actual LAN hardware. Is the file server secure? Could someone easily remove the file server from the area? Providing access to hardware compromises security.

There are other aspects of security to consider, but the best method to effectively secure a LAN is through proper network management. The question of security addresses the safety of the data and applications. Is the local area network open to a virus or any other program that might harm the data and applications on a local area network? Is the data safe? If data is accidentally erased, can it be recreated? This is done by storing backup copies of the network's files. Is there a disaster recovery plan? The file server's storage may be backed up on tape or some other medium, but a backup is of no benefit when the building burns. A copy of the data should be kept off-site in a secure place.

The evidence is quite clear that a security study should be done. The level and degree of security required for a local area network must be defined. Once defined, it must be incorporated into the design of the system.

Proper System Management

System management can be divided into two categories:

1. Procedures and policy
2. Maintenance

Procedures and Policies

A few local area network managerial policies and procedures were discussed as they related to the LAN's security, such as making sure that servers and other hardware are inaccessible, preventing viruses, creating file backups, and having a plan for disaster recovery.

Other procedures must be established to ensure a smooth and efficiently operating LAN. The following are a few procedural strategies you might consider, but they are by no means all the procedures and policies you might care to implement:

Password Schemes: User password schemes are an important aspect of security. A procedure needs to be developed to create user passwords and record them in a secure place in case access is required in an emergency. Procedures should also be created to change passwords at regular intervals, and to remove passwords for users who no longer have authorized access to the LAN, such as terminated employees.

Data Archiving: Data backups are a crucial part of LAN maintenance. A procedure needs to be established that defines when backups are performed (usually at night when users have logged off the system), how often they are performed, and how backup tapes or disks should be rotated and stored.

User Training: A procedure to train new users should be established. Such training can be performed by the network administration staff, or department managers or staff members can be designated as LAN trainers. User training manuals explaining how this specific LAN environment functions could also be valuable and save a lot of frantic user phone calls.

Policies for Application Programming: A procedure needs to be established to create batch files and support for new users. In addition, rights for access to different LAN databases have to be considered. This is particularly true where LAN-based data should be kept confidential, as would be the case with payroll records. Policies for assigning rights to applications need to be created to assign read and write authority to some managers while providing read-only access and write-only to other users. There are other applications-oriented policies to be considered, but these will vary depending on the network environment.

Connections to Other Systems: Interconnecting departmental work groups or even connecting LANs to other LANs or LANs to wide area networks (WANs) or mainframes requires its own unique considerations.

Network Logs and Journals: All the information about the local area network installation has to be recorded. Even if there is only one LAN administrator, that individual cannot possibly remember every configuration option or every user password. And if there is more than one person administrating the network, this problem is compounded. Having a master notebook or file of network information can be invaluable when it comes to maintenance. This log should contain every detail about LAN hardware and software configuration, and the types of equipment and connections used. Records should also be kept of various part numbers and warranties in case equipment or software needs to be replaced or upgraded.

Assigning New Users: Procedures must be developed for adding users to the system. These include the files and applications to which that user should have access, assigning an electronic mailbox, training, etc.

Deleting Old Users: When users are removed from the system, a procedure should be used to delete that user's password, remove his or her mailbox, archive that user's files, etc.

Installing Software: When new software is added to the network, it has to be installed on the hardware that will make it accessible to the users who need it most. Is there enough capacity on that hardware to handle another application and its associated data? Are the appropriate printers and devices available to support the application, such as a laser printer for a new word processing package? Also, users must be notified and trained in using the software.

Planning for Network Growth: In designing the network, areas of growth should be identified and the LAN configured accordingly. For example, if the organization's sales staff is expected to double within the next year and each new employee requires a workstation, procedures should be established to accommodate that growth on an ongoing basis. After all, stringing new cable and installing a number of new users cannot be accomplished overnight. Further, the hardware and software associated with those growth areas should be able to handle the added capacity.

Network Printing: In setting up the network, printers should be strategically located to accommodate those users who need printing services the most. For example, those departments that generate large amounts of correspondence should share a laser printer, but other users may need occasional access to that same printer.

Network Monitoring: A procedure to monitor network performance on an ongoing basis can help prevent trouble before it starts. This can be done using sophisticated network management hardware, like a protocol analyzer, or even by running periodic performance tests at various points on the network.

Maintenance: Policies and procedures must be established for both overall maintenance and day-to-day management of a local area network. System maintenance functions can be grouped into two major categories:

- System Hardware Maintenance: Once the hardware is installed there are some procedures to follow to make sure it continues to run properly. The hardware should be tested periodically to make sure it is running at or near specification. Diagnostic software is available that will provide a benchmark for performance.

Of course, if hardware fails, it will need to be repaired or replaced. Again, diagnostic software and troubleshooting devices can be used to isolate the problem and determine which hardware component needs to be replaced.

Upgrades should also be considered part of the hardware maintenance process. For example, if the amount of network data has grown to fill the server's disk capacity, a larger disk will be required. Or if there is so much data stored on the network that it cannot be backed up in a timely manner, it might be time to upgrade to a faster backup system.

No matter what kind of hardware maintenance is required, it is important to keep a written record of all hardware repairs and upgrades. Keep a record of all changes to the network, including serial numbers, diagrams of hardware configuration, etc.

- System Software Maintenance: Once the application software has been installed, it will require little maintenance, if any. However, it may be necessary to upgrade that software as new versions become available. This will mean keeping a record of all the software packages installed on the network, including application version numbers and serial numbers. Upgrades for certain application packages are provided free to qualified users, so it is important to keep receipts to verify purchase dates and other information.

Part of software maintenance is providing applications to the users who need them. This will mean configuring the system, writing batch files, and modifying access programs to accommodate the needs of different departments and users. It will also mean changing the configuration to accommodate new users. However, be careful when modifying software configurations. Modifying a program to suit the needs of one user or department, can affect the other 50 users on the network.

One aspect of software maintenance that is often overlooked is keeping archived copies of the software as well as the data those applications support. If the server ever fails or "crashes," it will be necessary to start from scratch. This will be a lot easier if all of the application software is at hand, including copies of the batch files and configuration files.

Again, be sure there is a written record describing all aspects of software configuration.

Consult With the Users

At this point in the design phase, it is time to categorize and prioritize user requirements. In the past, it was a traditional practice to develop the network without consulting the users. But without ongoing consultation, a LAN may be developed that is technically excellent but fails to meet the needs of the users. If the users needs are not met, it does not matter how much money and technology was put into the LAN, it is a failure. Therefore, never leave the users out of the system life cycle, and be sure to document the information as you accumulate it.

User contact makes it easier to select the right components and create a design that can grow with the needs of the organization. Remember, too, that users may be somewhat suspect of a local area network. By staying in contact with the users, their fears and apprehensions can be lessened. The attitude most users have to change becomes favorable when they realize that what they say and feel about the system is taken seriously. And as users become more knowledgeable about the LAN, they tend to offer more solid information about what they need now and in the future.

What users might need in the future is very important. The network must not be designed to meet only present user and organization's requirements. The price for that kind of thinking will have to be paid later, and it can be costly.

So, above all, consult with the users. Listen to the users. Try to remove their apprehension. Often, apprehension surfaces as negative attitudes. Try to instill a genuine feeling that the LAN is not something that will remove their jobs, but something that will help make their jobs easier. Again, learn from the past. It is well worth it.

Development of Procedures Design

When any new system is introduced to an organization, the methods used by that organization may need to be modified or eliminated. Many times new procedures need to be developed to complement the smooth operation of the network. Be prepared to modify the procedures or methods used to perform tasks affected by the network. In creating a new LAN, it is important to be able to design new methods of operation for the organization, and to be able to defend those new methods.

Development of Information Flow

From the study phase, information was collected on the organization on a broad scale. Part of this information came from the background of the organization. However, when designing a local area network, not only may procedures change, but organizational functions may change as well. The flow of information will probably change as well.

As an example, if the warehouse foreman was not receiving daily reports, it would be harder for him or her to perform. This was identified as a problem during the study phase. The local area network should be designed to remedy that particular situation, so the flow of information is changed.

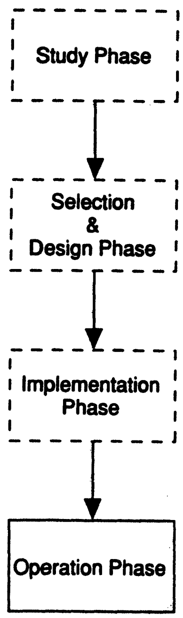
In designing the network, it may be necessary to dramatically change the overall flow of information, or the information flow may not have to be modified at all. The design of the network should accommodate a change in the information flow when it remedies a problem situation. The idea is not to change what is already in place just for the sake of changing it. Data flow and design must serve a specific purpose.

Select the Best Topology

Selecting a network topology is an extremely important point, and one that must not be overlooked in the design process. In many cases, selection of the media access method, i.e., Ethernet, Token Ring, etc., determines the topology, however this is not always the case.

Linear bus networks, primarily Ethernet and occasionally ARCnet, work well in laboratory and school environments, where machines are to be placed side-by-side. Distributed star topologies, such as those used by the ARCnet and 10BASE-T standards, are an excellent choice since they lend themselves well to troubleshooting and expansion. If a Token Ring network is chosen, the topology can

Figure 5.
Hardware and Software Implementation



only be a distributed star. Token Ring networks are internally a ring although externally they resemble a distributed star.

Select the Appropriate Transmission Media

There is a little more flexibility when choosing a transmission medium. Coaxial cable (for Ethernet and ARCnet) has an incredibly large installed base and is still being widely used, although more networks are realizing the benefits of twisted-pair wiring, which is what most Token Ring networks use. Twisted pair is a much more versatile and inexpensive medium, and is capable of high transmission rates. For Ethernet and ARCnet networks, twisted pair is an excellent choice, and for Token Ring networks it is your only choice.

In many cases, the choice is to go with shielded or unshielded wiring. Shielded wiring accommodates longer cabling distances and makes higher transmission rates easier, although it is more costly.

Fiber-optic cable is another option, and is available for most networks. Users of this expensive and fragile medium usually require a high level of security and/or need to cover long distances with high bandwidth. Many networks consist of fiber-optic cable for vertical and long distance runs, and twisted pair for cable runs from distribution closets to the desktop. In Ethernet networks, thick coaxial may be used for vertical or long distance runs.

Evaluate Software

It is the LAN developer's responsibility to evaluate the different software packages available and select those that most closely suit the requirements of the users. The users' software application needs were determined in the study phase. Now the right software must be identified to meet those needs.

There are many considerations that must be kept in perspective when evaluating software. How closely does the software meet the needs of the users? Is the software considered easy to use and does it have many bells and whistles? Remember, if it does not meet the users' needs, then no matter how user friendly or how many bells and whistles, it is the wrong choice.

If security is a key item, then the software package must meet the requirements for security set forth by the organization. However, the LAN design may compensate for security not offered in the software package. And what about hardware requirements? Is the LAN hardware powerful enough or appropriate for the software being considered?

Software maintenance and documentation are also crucial. If bugs are found in the software, does the vendor agree to correct these problems? Is the software too expensive? Are updates to the software too expensive? Does the documentation provide enough clear and succinct information to make it easy to learn the software? Is training required to learn this software? If so, then how much does the training cost? Is the software compatible with the network to be installed?

Once all of these factors have been taken into account, an authoritative recommendation can be made for the right software package that will suit the users' requirements.

Evaluate Hardware

The hardware that comprises the local area network must also be evaluated. After all, the hardware selected must support the software that was recommended. Consider the maintenance contracts for the hardware. Is the supplier or installer going to supply ongoing hardware support? Does the vendor guarantee the hardware for a period of time? Should a hardware maintenance service contract be signed?

The hardware selected also should comply with industry standards. If a piece of hardware is used that is not well known or well accepted in the computer industry, the organization could be in trouble when problems develop. The documentation of the hardware should be adequate. And is there training needed to operate the hardware? If so, is the training readily available, either from the vendor or the supplier?

Finally the price of the hardware should not be overlooked. You do not always "get what you pay for." Many times an organization was greatly overcharged for the hardware they purchased. Scrimping on hardware when installing or upgrading a LAN is also not always advisable. Be certain to invest in a hardware platform that has enough flexibility to suit the organization's needs, and that can handle future requirements.

Create the Most Appropriate LAN

Creating the most appropriate local area network for the environment requires all the information gathered in the study phase and presented in the feasibility report. In addition, information on the evaluation of hardware and software will assist in the design process.

The problems identified during the study phase should be restated. Then a system can be designed that addresses and corrects each of the problems listed. In addition the system must meet the requirements set forth in the hardware and software evaluations. It is also important to be able to make alterations to the design, and to be able to consider alternative products or methods.

Prepare a Design Phase Report

Once the system has been designed, the design should be presented in a design phase report. This report is reviewed by the committee of those involved with this project. If the design is acceptable, then the next phase, implementation, begins. If the design is not suitable, then another design must be created that addresses the objections raised by the committee.

The Implementation Phase

Once the design of the local area network has been approved, it's time to implement the hardware and software (see Figure 5). The implementation phase or development phase as it is sometimes called, is where the local area network is actually delivered, installed, tested, and made operational.

This is the point where the users actually begin to see the system. The days and weeks of information gathering and analysis are now showing tangible results. The workers' everyday routine will also begin to change during system implementation so the installer, the person responsible for the actual implementation, must be aware of the users and their feelings. The wise system implementor takes the time to talk with the users, to explain what is occurring, and to help them develop a positive attitude toward the network.

This is the stage where the LAN starts to become a reality in the minds of the users, since more people will be seen at work in this phase than in the previous study and design phases. A few people can gather the information and design the system. It requires more people to actually install the system and ensure that the system works correctly and accurately. So with all this visible activity, it is best to try to gain the users understanding and cooperation.

It should be noted that the components designated during the design of the LAN may be modified at this point. Considerations that were overlooked in the study and design phase can be corrected in this phase. It should also be noted that any major redesign or change in key pieces of hardware or software must be approved by the review committee before they are implemented.

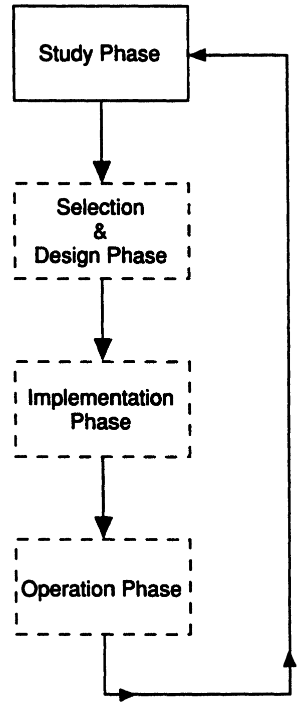
The other two phases could be conducted according to a defined set of actions or activities, and the same should hold true for the implementation phase (see Figure 6). The major activities or steps in the implementation phase are:

1. Implementation plan
2. Computer program design
3. Review meeting
4. Equipment installation
5. Computer program development
6. System test
7. Software test
8. Manuals developed
9. Personnel training
10. Final review meeting

It is not uncommon for the implementation phase to take substantially more time than the study or design phase. A great amount of work must be accomplished in this phase before it can be called complete. The amount of the activities listed above suggests that more time has to be allotted for this phase, and this will hold true in most system installations.

The timetable for LAN installation may vary. It could take a couple of weeks or longer, depending on the scope of the project. It is possible to have the hardware working properly in one month, but the actual application software may take longer to install and integrate. The amount of time spent on each activity is determined by the scope of the project.

Figure 6.
The Implementation Phase



Implementation Plan

Once the system has been selected and the specifications detailed, the next step is to develop a plan to describe how this system is to be installed. The implementation plan carefully outlines on paper the procedures and sequences necessary to enable the system, providing a blueprint that will result in an operational, real-life system.

An analogy would be that of a couple building a house. The couple finds a location they desire and then consults with the builders and developers—the study phase. Once the location has been defined, the couple become more specific about the type of house they want to build. The number of bedrooms needed is determined, the types of windows are selected, and many more details are worked out regarding the house. The builder designs a house and produces blueprints that the couple can approve—the design phase.

After the design of the house has been approved, the builder does not just go out and start digging foundations, erecting walls, and pouring concrete. The contractor must have a plan to schedule these tasks. By defining and following a series of assigned tasks in a specified sequence, the result is a livable house that can be built for a predictable cost. The same process can be applied to implementing a local area network. A plan must be established and followed to develop the system successfully. Depending on the size of the project, the implementation plan may be quite complex or relatively simple. However, a good implementation plan considers all the other activities listed above, and must allow for flexibility.

Those who develop the implementation plan should have competency and experience in system development, and should be well-versed with local area networks. After all, how can someone with limited experience of local area networks be expected to develop a plan to implement a LAN? This occurs more times than it should, and the organization managers should be aware of this.

If a specific vendor is supplying the hardware and packaged software, that firm may also be contracted to do the installation. Be certain that the vendor is competent and experienced. It might be advisable to use a consultant to oversee the installation, but one vendor may still provide the hardware and packaged software.

Computer Program Design

The computer programs that have to be considered in this phase are both applications that need to be customized or developed, and commercial programs selected during system design.

Customized software development may also be necessary. This process is not always handled by an internal network developer, but could be assigned to an independent software development company. There is a good number of qualified programmers and programmers/analysts in the marketplace, and these programmers are quite capable of designing and implementing custom applications. Before hiring a software developer, investigate whether the program designer responsible for designing the applications is competent and knows the differences between coding programs for single-users versus multi-user environments.

Review Meeting

A review meeting was held at the end of both the study and design phase. There should be an intermediate review meeting as part of the implementation phase as well. This review meeting should include the users and/or persons from the organization who are involved with the local area network project. The purpose of this meeting is to determine if the implementation plan is acceptable to the users and those persons of the organization connected with the LAN project.

Two actions may result from this review meeting. First, if the implementation plan is not acceptable, it must be modified or totally redesigned. If the plan is changed, the implementation plan would again go before the review meeting after the modifications have been drafted. Second, if the implementation plan is approved, installation of the system continues according to the implementation plan.

Equipment Installation

As stated earlier, the equipment vendor may be contracted to install the equipment, or a third party could do the installation. There are a number of companies that do not do system design work but whose sole business is installing computer systems, performing the installation according to the implementation plan. No matter who puts in the hardware, the installation should be supervised. Many times, installation is supervised by the person who developed the implementation plan.

Once the equipment is installed, it must be checked and rechecked. Many systems have failed soon after installation was completed because the equipment was not verified to be truly functional after it was installed. Testing equipment eliminates the pointing fingers and accusations later.

Computer Program Development

In those environments that require customized applications, the application programs can be started while the equipment is being installed. In fact, they can be started once the review committee gives its approval, since it is not necessary to have a functional LAN to write the program code.

One person should be designated to monitor the development of the application programs. This may be a project leader who in turn reports to the project manager. In any event the person who oversees the application program development must be familiar with application design and implementation. This person must also know programming methodology and techniques.

System Test

Just as each component must be tested when it is installed, the entire system should be tested. Each individual piece of equipment was tested to be functional, so now it is time to make sure that the entire system is functional and the pieces work together as a whole. These tests are run and rerun using different variations. Hopefully, any bugs in the system will be uncovered and corrected at this time.

Software Test

Once the system hardware is up and running, the application programs can be loaded unto the LAN. These applications must undergo close scrutiny to ensure their design integrity. It is better to take some time in the beginning to make sure the programs are correct (if customized) or correctly installed (if they are off-the-shelf applications) than to take much more time correcting them later. If the applications do have to be changed later, not only will the programs have to be modified, but so will the data. So using extra care in installing and testing the software will help maintain data integrity. The best LAN design will be of no use if the data is corrupt.

Manuals Developed

Manuals should be developed to document every facet of the system. The administrators' manual should include the cable layout—a cable map. This would include what type of cable was used, where repeaters, bridges, or gateways have been installed, and other details. The structure of the file server, including structure of the hard disk, should also be recorded in the manual. In fact, anything that has an impact on the design of the network should be in the manuals; they should be considered documentation for the system.

With proper documentation, it will be easier for a third party who may come onto the scene at a later time to understand the system. It will also provide a record to jog the installer's memory when it comes time to modify or repair the system.

All system documentation should be precise and to the point. The time spent on documentation or manuals during installation can save much time and money in the future.

Personnel Training

The local area network may be sophisticated enough to require user training. Training should be accounted for in the implementation plan. There may be two types of training needed:

1. Network administrator
2. Network user

One individual or more must take responsibility for the local area network. The network can not run itself day in and day out after its initial setup. There are various duties and monitoring that must take place to ensure that the LAN operates smoothly. Those who are responsible for the smooth operation of the LAN must be chosen and trained on the hardware used throughout the network.

It is also a good practice to train local area network users, although this does not have to be as extensive as the training for the network administrators. The users should know the basic principles behind local area network. This helps to ease their fear of the network. The users should also be trained in certain operations of the network operation system, jut as they are trained in the use of MS-DOS, so they can do their jobs effectively without having to continually ask questions of the network administrators.

Final Review Meeting

After the plans for training are established, a final review meeting is held. This meeting typically goes in one of two directions. If everything has goes according to plan, the implementation is satisfactory and the operation phase should soon begin. However, if the implementation is not satisfactory, the areas of contention should be defined and discussed.

Should the implementation be at fault, then the developer has the responsibility to make any corrections. There are other difficulties that arise as well. Too often, once the implementation is complete, the organization may want items included in the system that were not part of the initial study. These items should be addressed by the implementor. It should be noted by the developer that these items were not part of the original design, but the developer should investigate them to see if these changes can be accommodated. Some modifications may require the system to outlive its usefulness before they can be made. However, if at all possible, these changes should be made before the system changes over to ongoing operation.

The review meeting usually allows the project to go onto the last phase, operation. Many times, new ideas or system enhancements are requested once the LAN is fully functional and users begin to see its power and the possibilities networking offers when it comes to handling information.

The Operation Phase

The last phase in the life cycle of the computer system is entered into only after the successful development and implementation of the local area network. This is the operation phase. It is comprised of four main parts or categories:

- Changeover
- Routine Operation
- System performance evaluation
- System change

The operation phase is considered the longest part of the LAN's life cycle since this stage is actually the system's ongoing operation or "life." More specifically, when the system is actually in operation, at least one of the categories listed above is in effect.

Changeover

Normally, a period of transition is required to change from an old system to a new one. It is critical that the developer remain involved throughout this transition period. System changeover is the most critical period in the entire life cycle of the computer-based information system.

Changeover is usually a one-way process, and must result in a system that is operationally acceptable. Regardless of the planning for the changeover or attention, unforeseen incidents and problems are probably on the horizon. This can be an anxiety-ridden time for all those involved, and workers may encounter a great deal of inconvenience during the changeover. Care should be given to keep up the morale and positive attitudes of the workers.

There are different methods used for changeover. Three of the most common methods are:

1. Cold Conversion
2. Parallel Conversion
3. Phased Conversion

Cold Conversion

With cold conversion, the old system simply stops and the new one begins. This method will have the most severe impact on the workers or users, and it is best to apply this conversion method when the work force is light or off of work. Weekends and holidays are a good time for this type of conversion.

Cold conversion is the simplest and least expensive changeover method. It is also the riskiest. The biggest risk is if the new system does not operate as expected, which creates many new problems. This technique is not suitable for organizations that rely heavily upon their information, or so-called "mission-critical" applications.

If the changeover method selected is cold conversion, there should be a contingency plan. The contingency plan is an alternate routine in the event the new system fails altogether or is not functioning properly or adequately. Never go into a cold changeover "blind."

Parallel Conversion

The parallel conversion approach allows the old system to be changed over to the new system with some overlap. The new system is placed into operation at the same time the old system remains functioning, so both systems are in operation for a specified amount of time.

During this period, both systems can be compared to isolate errors that may occur with the new system. With parallel conversion, if errors should occur with the new system, the organization is not hurt since the old system is still functioning. An additional advantage to this approach is that the new system can be fine-tuned and made as free of bugs as possible.

The parallel conversion method is almost risk free. The major drawback to this conversion method is that significant resources must be maintained to have two systems operating simultaneously. It can become expensive to have two systems running side by side.

Phased Conversion

With the phased conversion method, conversion is done in modules or phases. The old system is converted to the new system in a series of pre-planned phases with specific timeframes. When one phase is completed successfully, the next phase begins. This process continues until all the phases have gone from the old to the new system.

The advantage to this approach is that if one phase encounters many problems or errors, the timetable to implement the subsequent phases can be adjusted. Also, correcting errors in an earlier phase can help smooth the transition of subsequent phases.

The phased conversion process allows the changeover to take place in several steps. In addition, this conversion method is less expensive than the parallel conversion method.

Routine Operation

When the changeover is completed, the local area network is considered operational. Unfortunately, all too often once an organization has spent considerable time and money to install a local area network, little consideration or resources are given to ongoing maintenance. If proper care is not given to the system, problems will arise that negate the benefits and savings of a local area network.

A computer-based information system should have maintenance routines in the following areas to keep the systems running at optimal efficiency:

- Hardware
- Software
- Programming

Hardware Maintenance

There should be clear, precise measures and procedures to keep the hardware in proper operating condition. Steps should be outlined for action to be taken when certain conditions arise.

For example, not all hardware repairs require the attention of a trained technician. If a printer does not print, it is a simple matter to check for a paper jam or see if it ran out of paper. Maybe the printer cables aren't intact. If the device continues to malfunction after routine inspections and checks, then a trained technician should be called.

This brings forth another issue regarding hardware maintenance. Each piece of hardware on the LAN should have a designated service agent. It should not be a task to find the correct names and telephone numbers of those vendors or dealers who support the hardware. In addition, a process called PM or "preventive maintenance" should be incorporated. Included under PM are the dusting and cleaning of equipment. Proper diagnostics should be run to ensure no problems arise. The cables should be checked to ensure that a good connection is maintained.

The bottom line is there should be a plan to maintain the hardware of the LAN. It may not be the best plan, but it is better than no plan at all. The plan can be perfected as the system becomes more familiar. Fear or anxiety over the hardware is no excuse for not having a maintenance plan.

Software Maintenance

The software and data must be stored on secondary media and these media must be maintained. If these programs or the data are lost, the organization can face large problems.

Backup procedures would fall under this category. The care given to floppy disks is another important measure to maintain software. Be sure to install and maintain the software using the procedures outlined by the distributor.

Software maintenance is no less important than hardware maintenance. In fact it could be of greater importance. A personal computer can be replaced, but can the data accumulated over the last six months be replaced should it be destroyed?

Programming Maintenance

For those systems that require customized applications, the software that was designed and created for the LAN must be maintained. These procedures require the same measures followed for software maintenance. In addition the custom-designed software will probably have to be modified periodically. These changes are needed to keep abreast of the changes in the organization, and they should be monitored very closely. Maintenance programming is not an issue to take lightly.

System Performance

After the new computer-based information system has functioned for a reasonable period of time, the system should be evaluated. The results of this evaluation are presented in an evaluation report that allows the organization to see if their local area network is providing the expected results.

At this time, the weaknesses and strengths of the system are brought to attention of organization executives. This is also the time that corrections can be made to the system. These corrections can have far-reaching effects when enhancements are added to the system at some later time.

Below are some of the items that should be addressed when evaluating the system:

- Cost Analysis
- Ease of Information Retrieval
- Data Integrity
- Personnel in contact with the system
- Amount of data processed
- Security
- Maintenance

System Change

All organizations are dynamic; they do undergo change. This is a fact that must be realized when implementing any computer-based information system.

During the evolution of the organization, it may be necessary to make changes in the network. Requests for change should be drafted for review. The request for change may trigger a response that ranges from a brief analysis of the request to an extensive investigation. This investigation may cause a return to an early point in the life cycle, requiring a complete redesign. An investigation could mean a return to the study phase. A simple request for a modification may yield a greatly modified system.

Just as the organization is dynamic and ever changing, so is a computer-based information system or a local area network. Many system analysts say that a system is never completed but always 90% complete. The system is constantly being updated and modified. ■

Financial Analysis: Downsizing to LANs

In this report:

Datapro Summary

As an information systems manager, you have decided that the time is right to begin a downsizing project in your organization. You know that taking some pressure off of the company mainframe will save money in the long run, but how do you sell the idea to others? By following some basic financial analysis guidelines, you will be better equipped to support your vision.

Do you know people like this? They have great ideas and a clear understanding of how technology is changing and how it should alter the strategic direction of their organization. Yet they are stymied when putting these ideas into motion because they cannot put a dollars-and-cents value on their visions. Boards of directors and others who control the purse strings are usually grounded in matters more tangible than vision. The result is that the long-range thinker's ideas never get put into action, and the organization loses opportunities to change and grow.

The fact is, vision is not enough. You must show the business benefit of every dollar spent. It's not enough to say that all benefits are intangible or intuitive. If you want your plan to fly, you have to show how it will affect the dollars flowing in and out of the company. You need a tool to supplement vision. You need financial analysis.

Okay, I know what you're thinking: This is going to be too complicated to understand, and I hate math. Relax. We all have access to computer spreadsheets and calculators that make it unnecessary to learn how to compute the numbers. We only have to understand some basic financial terms and how to use an automated tool for the number crunching. The important thing is knowing how to interpret the numbers.

To begin, I'll discuss basic financial concepts and some common, practical types of financial analysis. Then, I'll look at an example that applies financial analysis to a common, important project for many of us—downsizing. Suggestions will be made for identifying costs and savings elements that should be included in the analysis. Finally, I'll explain the value of post-cost analysis and how you can use it to get things rolling in your organization. While not a complete, academic treatment of financial analysis, this report will give you an understanding of the basics and get you on your way to using financial analysis techniques to sell and support your projects.

The Financial Concepts

Before you can attempt the financial analysis of any project, you need to understand the financial concepts. Let's take a minute to consider a few key ideas.

Cash Flow

The purpose of the financial analysis is to establish as closely as possible how a project will affect the dollars flowing in and out of the corporate pocket. The typical project initially requires spending money to fund it—that is, dollars flowing out of the company. In subsequent years, dollars flow back into the company as savings (or reduced costs) from the project. We have to

This Datapro report is a reprint of "Speaking in Numbers" by Marc Dodge, pp. 57-66, from *LAN Technology*, Volume 7, Number 8, August 1991.

determine, for each year of the project, how much money is moving and in which direction. Cash flow is money moving in and out of the corporate pocket.

Analysis Period

An analysis period is the number of years that the project will affect the cash flow of the organization. Let's think this out. Typical useful lives for components in a computer systems project are three years for the software, five years for the hardware, and 10 to 15 years for the wiring. We should select the number of years in the analysis period by considering the useful lives for components and the tax laws. Choosing an analysis period has to be tempered with the fact that estimates of cash flows become very shaky beyond five years, especially for projects involving computer technology. Three to five years is a good rule of thumb.

Time Value of Money

If you loan me $100, do you care whether I pay it back tomorrow or next year? Of course you do. If I keep the money for a year, you cannot spend it. You also lose the opportunity to put the money in a bank and earn interest. The time value of money is based on the understanding that a dollar in hand is worth more than a dollar received in the future. We have to account for the time value of money in our analysis.

Consider a project that requires a $10 investment today and returns $5 in each of the next two years. The savings exactly equals the investment. Is it a wash? That's right, it's not a wash. The $10 flows out of our pockets today. We receive the first $5 after 12 months and another $5 after 24 months. To determine whether to make the investment, we have to account for the timing of cash flows in our analysis. In financial parlance this is called discounting. In plain terms, we can think of it as adjusting the value of future cash flows, so they are stated in terms of "today's dollars." This allows us to compare apples to apples in the sense that the dollars in each year of the project analysis period are presented in current-year dollars. This enables us to compare competing projects directly and equally.

Discount Rate, Cost of Capital, and Rate of Return

The first two terms refer to the cost of money for an organization. While they are often used interchangeably, the cost of capital is a method of determining a discount rate (the rate at which money is discounted). Companies usually compute the cost of capital by considering how the company is financed. Often, this is a mix of common stock, preferred stock, retained earnings, and the cost of borrowing money. Other companies consider the returns of alternative projects instead of the cost of capital.

In any case, we don't have to worry about this computation. The easiest thing to do is check with your accounting department. They will tell you the appropriate discount rate in the financial analysis. The rate of return is the interest rate that expresses a return on the initial investment and subsequent savings over the analysis period. For example, my company has a published goal of achieving a certain rate of return on investments over the next five years. This target makes it a logical rate of return to use for evaluating projects.

Taxes

The federal government receives 34% of corporate income. Naturally, this plays a major role in financial analysis. Uncle Sam is a big partner in the business, since "he" gets 34 cents out of each dollar earned. Understandably, companies are allowed to deduct the cost (or expenses) of producing income.

Depreciation

We invest money in computers and other business assets to make money. Let's say we buy an 80486 microcomputer to run part of our business. The computer will not be useful forever. The tax laws allow us to deduct a portion of the computer's value from each year's gross income. The laws recognize that the computer is "consumed" and that the expense was made so the company could earn taxable income. Depreciation deductions reduce total income, so the larger the depreciation deduction, the less tax the corporation pays. Depreciation is very important to the financial analysis.

The tax law provides six recovery periods for property: three, five, seven, 10, 15, and 20 years. For example, five-year property includes computers, copiers, and computer-based telephone office switching equipment. Seven-year property includes office furniture and fixtures, and 15-year property includes equipment used by non-telephone companies for voice and data communications.

Fortunately, depreciation calculations are not difficult. Most spreadsheet programs have functions for calculating the different types of depreciation. The best idea is to get your accounting department staff to help with the math. Just as important, they will explain company-specific guidelines for applying depreciation law.

Financial Analysis Tools

With a basic understanding of the key financial concepts, we can now look at a few analysis tools. There are several ways to analyze a project, but the tools below are two of the most common methods available.

Present Value and Net Present Value

The first commonly used method of financial analysis is present value (PV). To understand PV, we will combine the concepts of discount rate and time value of money and return to our time value of money example.

Our analysis period is two years and our accounting department told us that the current discount rate for the company is 10%. We are going to invest $10 at the start of 1991, and we predict that this will result in savings of $5 at the end of 1991 and another $5 at the end of 1992.

1991 Initial cost	$10
1991 Savings	$ 5
1992 Savings	$ 5

Present value formulas provide a method for discounting cash flows and restating those cash flows in equivalent values at a common point in time to account for the time value of money. The $5 received at the end of 1992 is not as valuable as $5 in our hands today. We will skip the actual formulas since they take a long time to work out and are available in spreadsheet programs. (If you are interested in the formulas, see the sidebar "The Math Behind Present Value Calculations.") Instead, we'll pretend that we used a financial calculator or computer spreadsheet to do the number crunching.

After discounting, the present value of the cash outflows is $10—the amount we invested at the start of the year. The present value of the cash inflows is $8.68. We converted the savings to be received at the end of 1991 and 1992 to today's dollars. We established what the present value of each cash flow is.

Net present value (NPV) is the total cash inflows less the total cash outflows. If the NPV is less than zero, the project is undesirable because it costs more than it brings in. If the NPV is positive, the project is desirable because it earns more than it costs.

In our example, the NPV is the $8.68 savings less the $10 investment, or negative $1.32. If money costs 10%, this project would be rejected because it does not even generate enough savings to pay for the principal and interest on the loan.

Most of us will use a computer to do the actual calculations. Even financial experts do not work with the formulas. However, it is important to understand the idea of discounting and the time value of money. You must also be able to interpret and explain the results. We will look at a spreadsheet example later.

Payback Period

Calculating the payback period is another common method of financial analysis. The payback period is the number of years it takes to recoup the initial investment for a project. It is easy to calculate and understand, so it is also very popular. If we spend $300 at the start of 1991 and get back $100 in each of the three subsequent years, the payback period is three years.

If the math is this easy, why bother with present value? There are several potential problems with payback. It does not usually consider the time value of money, and it does not consider the cash flows after the initial payback is established. (If you were to factor the time value of money into your payback analysis, you might as well use the NPV method.) Payback can also incorrectly rank competing projects in order of economic value. This is best shown with an example:

	Project A	**Project B**
Initial Cost	$300	$300
Savings 1991	$100	$ 0
Savings 1992	$100	$ 0
Savings 1993	$100	$300
Payback period	3 years	3 years

Both projects cost $300 and return $300 in savings over the three-year analysis period. Payback period analysis shows that the two projects are equally valuable, but they aren't really. Project A returns money more quickly than project B. Considering the time value of money, project A is the better choice. Also, the payback does not tell us anything about cash flow in the years after 1993. The future cash flow may completely change the picture.

Although payback is not a tool you want to stake your career on, often a single project will be so favorable (with a very short payback) that a detailed financial analysis using present value is overkill. Some LANs have a payback period of only a few months and a dramatic impact through increased productivity and labor and maintenance savings. If the payback is under a year, it is usually pointless to do a detailed financial analysis. It will not provide additional, meaningful information.

The Math Behind Present Value Calculations

Computers and calculators shield us from the math of financial analysis. However, the math is not difficult. The example below shows an investment of $10 followed by a $5-per-year savings for the next two years.

The formula is

$$\text{net present value} = \frac{CFn}{(1 + i)^n}$$

In this formula, CFn = cash flow in year n (the current year is year 0), i = discount rate, and ^{n} = analysis period year (think of it as years from the current year).

	Cash Out	**Cash In**	
	1991	**1992**	**1993**
NPV =	$-10.00	+5	+5
	$(1+.10)^0$	$(1+.10)^1$	$(1+.10)^2$
NPV =	-10.00	+4.55	+4.13
NPV =	-10.00		+8.68
NPV =	-1.32		

Putting It All Together: A Downsizing Example

Let's bring our previous discussions together with an example—downsizing. A typical downsizing project involves comparing the costs of moving and operating a mainframe application on a less expensive computing platform. We have two alternatives: install a LAN to run the application or continue to run the application on the mainframe.

We must consider the money that will go in and out of the corporate pocket for each alternative. The costs and benefits of a project can be classified as tangible (recurring and nonrecurring) and intangible (recurring). The first step is to describe the possible costs and savings for each category.

This part of the financial analysis involves creative thinking and an understanding of the business. It is also the part that stymies people more than the math. If you leave out a relevant savings or cost, it does not matter if 10 pages of numbers back up the calculations. It's still wrong and the project will suffer a serious setback and loss of credibility. Take your time with this part of the analysis and check your thinking with objective co-workers. Below are common items to consider for computer-related projects.

Tangible costs or savings (non-recurring) include the following:

- Application development
- Hardware
- Operating system software, including network operating system
- Initial training
- Initial wiring
- Corporate network redesign to support expanded connectivity
- Sales tax
- Installation labor

Tangible costs or savings (recurring) include the following:

- Repetitive training for end users and management
- Hardware maintenance
- Software maintenance for the applications, operating system, and network operating system
- Application support
- Wiring changes
- Local and corporate-wide network management
- Security
- Help desk support
- Software distribution
- Telecommunications
- Productivity and labor

Intangible costs or savings (recurring) include the following:

- Operating flexibility and adaptability
- Employee morale and job satisfaction
- Change in corporate image
- Comfort zone of decision makers
- Processing power at end-user workstations
- Application development cycle
- Ability to manage the department and network

No matter how sophisticated or complete the analysis is, it is only as good as the numbers put into it. Don't forget to weigh the labor costs or savings associated with your end users by including their benefits. As a rule of thumb, benefits are about one-third of the base salary. For many projects, the costs of labor will far outweigh the costs of hardware and software, so be sure to use a realistic number.

Intangibles deserve a special word. Some of the most important reasons for supporting one alternative vs. another may revolve around intangible items. Do your best to put a dollar amount on these items. Decision makers are much more likely to be swayed by reasonable, fair estimates.

Network administrators frequently have trouble selling information technology projects to upper management. Managers resist spending money on computer equipment since it seems unnecessary the way most projects are presented. However, the cost of a typical microcomputer represents just a few weeks of a middle manager's fully loaded salary. Find the productivity gains, place a cost on them, and your project will fly.

The Analysis of Downsizing

Figure 1 shows the operating costs stated in terms of cash flows for the two alternatives. Productivity is measured in end-user labor costs, and the associated productivity costs include user salaries, benefits, and payroll taxes. Our analysis period is six years, which allows the LAN hardware to be depreciated completely. Costs have been worked out for each year of the analysis period for training, hardware and software maintenance, and productivity. This is just a sampling of possible cost items that could be part of your own analysis.

Figure 1.
Dollars Spent for Each Alternative.

		Alternative #1: LAN			Alternative #2: Mainframe		
Year	Cost Description	Cost	Inflation Rate %	Inflated Cost	Cost	Inflation Rate %	Inflated Cost
1991	Training	10,000	5	10,000	0	5	0
	Maintenance	4,000	5	4,000	8,000	5	8,000
	Productivity	0	5	0	20,000	5	20,000
1992	Training	5,000	5	5,250	0	5	0
	Maintenance	4,000	5	4,200	8,000	5	8,400
	Productivity	0	5	0	30,000	5	31,500
1993	Training	0	5	0	0	5	0
	Maintenance	4,000	5	4,410	8,000	5	8,820
	Productivity	0	5	0	30,000	5	33,075
1994	Training	0	5	0	0	5	0
	Maintenance	4,000	5	4,631	8,000	5	9,261
	Productivity	0	5	0	30,000	5	34,729
1995	Training	0	5	0	0	5	0
	Maintenance	4,000	5	4,862	8,000	5	9,724
	Productivity	0	5	0	30,000	5	36,465
1996	Training	0	5	0	0	5	0
	Maintenance	4,000	5	5,105	8,000	5	10,210
	Productivity	0	5	0	30,000	5	38,288

Each cost was adjusted for inflation at five percent per year. Instead of discounting the money, we're increasing the operating costs. These adjustments were made because we determined the costs in 1991 dollars. These costs will rise in subsequent years like everything else, and we must account for it in the analysis.

By the way, this is very common type of business analysis. The cost of alternatives are compared to see which will cost the least. It's not a matter of how much money they save or whether they are worthwhile. We assume that one alternative must be chosen. Imagine that the computer application for which we are choosing a computing platform is used to track billing and accounts receivable for a company. It is apparent that the application is necessary for the business to operate. The question is which alternative can do the job for the least amount of money.

Again, figuring operating costs is just one of several steps in performing financial analysis. You are essentially compiling information to use in a later step. Therefore, while it gives you a thumbnail sketch of a specific project's feasibility, it's not an independent financial analysis tool.

After compiling the costs (including depreciation) for each alternative for each year of the analysis period, we are ready to use a spreadsheet to do the financial analysis. For the proposed project, we usually have to calculate one of the following: net present value, which is a dollar value; payback period, which is usually expressed in years; or rate of return, which is expressed as a percentage. We will use a Lotus Development Corp. Lotus 1-2-3 spreadsheet to do the calculations. Figure 2 shows the spreadsheet. The numbers you will need to provide are shaded. All other numbers are computed by the spreadsheet. (The Lotus formulas I used are listed in Figure 3.

"Before tax operating cost" is calculated from Figure 1. "Computer depreciation" is the depreciation allowed in each year for the purchase of the LAN hardware. Your accounting department will help you calculate depreciation allowance. The "Total after tax cash flow" is the sum of "Computer investment" and "After tax operating cost" less "After tax depreciation."

The "Total after tax cash flow" for the LAN is $61,022; for the mainframe it is $163,992. (While cash flow numbers are generally expressed as positive figures, they are understood to be positive or negative—inflow or outflow, respectively—by each spreadsheet column's definition.)

Present Value, Payback, and Rate of Return

The next step is to account for the time value of money by discounting the after tax cash flows (see Figure 4). There are two ways to discount these numbers. The first way is to bring the "Total after tax cash flow" column in Figure 2 forward to column B in Figure 4. Then calculate the net present value using Lotus' @NPV function. The second way is to individually discount each year's "Total after tax cash flow" and enter the resulting net present values in Figure 4, column D. The discount rate of 10% is either programmed into Lotus or used in the equation described

Figure 2.
Conversion of Cash Flows to After Tax Figures

	A	B	C	D	E	F	G
1	Alternative #1: Install a LAN to run the application.						
2							
3							
4		Before Tax	After Tax	Computer	After Tax	Computer	Total After
5		Operating Cost	Operating Cost	Depreciation	Depreciation	Investment	Tax Cash Flow
6							
7	1991	$14,000	$9,240	$10,000	$3,400	$50,000	$55,840
8	1992	9,450	6,237	16,000	5,440		797
9	1993	4,410	2,911	9,600	3,264		(353)
10	1994	4,631	3,056	5,750	1,955		1,101
11	1995	4,862	3,209	5,750	1,955		1,254
12	1996	5,105	3,369	2,900	986		2,383
13							
14	**Totals**	**$42,458**	**$28,022**	**$50,000**	**$17,000**	**$50,000**	**$61,022**
15							
16							
17							
18	Alternative #2: Continue to run the application on the mainframe.						
19							
20							
21		Before Tax	After Tax	Computer	After Tax	Computer	Total After
22		Operating Cost	Operating Cost	Depreciation	Depreciation	Investment	Tax Cash Flow
23							
24	1991	$28,000	$18,480		$0	$0	$18,480
25	1992	39,900	26,334		0		26,334
26	1993	41,895	27,651		0		27,651
27	1994	43,990	29,033		0		29,033
28	1995	46,189	30,485		0		30,485
29	1996	48,499	32,009		0		32,009
30							
31	**Totals**	**$248,473**	**$163,992**	**$0**	**$0**	**$0**	**$163,992**

This part of the spreadsheet converts the cash flows to after tax figures. The shaded areas indicate the numbers you need to provide.

Figure 3.
The Lotus 1-2-3 Formulas Used to Generate the Spreadsheet Numbers in This Report

```
C7:  +B7*0.66
E7:  +D7*0.34
G7:  +F7-E7+C7
C8:  +B8*0.66
E8:  +D8*0.34
G8:  +F8-E8+C8
C9:  +B9*0.66
E9:  +D9*0.34
G9:  +F9-E9+C9
C10: +B10*0.66
E10: +D10*0.34
G10: +F10-E10+C10
C11: +B11*0.66
E11: +D11*0.34
G11: +F11-E11+C11
C12: +B12*0.66
E12: +D12*0.34
G12: +F12-E12+C12
B14: @SUM(B7..B12)
C14: @SUM(C7..C12)
D14: @SUM(D7..D12)
E14: @SUM(E7..E12)
F14: @SUM(F7..F12)
G14: @SUM(G7..G12)
C24: +B24*0.66
E24: +D24*0.34
G24: +F24-E24+C24
C25: +B25*0.66
E25: +D25*0.34
G25: +F25-E25+C25
C26: +B26*0.66
E26: +D26*0.34
G26: +F26-E26+C26
C27: +B27*0.66
E27: +D27*0.34
G27: +F27-E27+C27
C28: +B28*0.66
E28: +D28*0.34
G28: +F28-E28+C28
C29: +B29*0.66
E29: +D29*0.34
G29: +F29-E29+C29
B31: @SUM(B24..B29)
C31: @SUM(C24..C29)
D31: @SUM(D24..D29)
E31: @SUM(E24..E29)
F31: @SUM(F24..F29)
G31: @SUM(G24..G29)
A35: +A1
B40: +$G$7
D40: +B40/((1+$D$38)^0)
B41: +$G$8
D41: +B41/((1+$D$38)^1)
B42: +$G9
D42: +B42/((1+$D$38)^2)
B43: +$G$10
D43: +B43/((1+$D$38)^3)
B44: +$G$11
D44: +B44/((1+$D$38)^4)
B45: +$G$12
D45: +B45/((1+$D$38)^5)
B47: @SUM(B40..B45)
D47: @SUM(D40..D45)
B49: +B40+@NPV($D$38,B41..B45)
A52: +A18
B57: +$G$24
D57: +57/((1+$D$55)^0)
B58: +$G$25
D58: +B58/((1+$D$55)^1)
B59: +$G$26
D59: +B59/((1+$D$55)^2)
B60: +$G$27
D60: +B60/((1+$D$55)^3)
B61: +$G$28
D61: +B61/((1+$D$55)^4)
B62: +$G$29
D62: +B62/((1+$D$55)^5)
B64: @SUM(B57..B62)
D64: @SUM(D57..D62)
B66: +B57+@NPV($D$38,B58..B62)
B75: +$G$7
D75: +$G$24
F75: +D75-B75
H75: +F75/((1+H73)^0)
B76: +$G$8
D76: $G$25
F76: +D76-B76
H76: +F76/((1+H73)^1)
B77: +$G$9
D77: +$G$26
F77: +D77-B77
H77: +F77/((1+H73)^2)
B78: +$G$10
D78: +$G$27
F78: +D78-B78
H78: +F78/((1+H73)^3)
B79: +$G$11
D79: +$G$28
F79: +D79-B79
H79: +F79/((1+H73)^4)
B80: +$G$12
D80: $G$29
F80: +D80-B80
H80: +F80/((1+H73)^5)
B82: @SUM(B75..B80)
D82: @SUM(D75..D80)
F82: +D82-B82
H82: @SUM(H75..H80)
```

in the sidebar. Although both calculation methods produce the same result, using Lotus is probably easier for most people.

The NPV for the LAN is $59,436. The NPV for the mainframe is $127,782. Because of the discounting process, these two numbers can now be compared directly to choose the best computing platform.

In this hypothetical cased, the total cost for the LAN is $68,346 less than the mainframe ($127,782 – $59,436 = $68,346). This means that over the entire analysis period from 1991 through 1996, we can run the application on a LAN for $68,346 less than on a mainframe. The financial analysis converted all cash flows to current year (1991) dollars. This allowed us to make an apples-to-apples comparison of the two bottom-line numbers.

Even with net present value calculated you may still want to know the payback period for the project (see Figure 5). The paycheck period shows how long it takes to recover the first year (1991) investment in LAN equipment. The initial investment is repaid from savings in operating costs compared with the mainframe.

The "Difference" column is the mainframe costs minus the LAN costs for each year. The difference in cash flows in 1991 is negative $37,360. This is mainly because of the $50,000 spent in 1991 for LAN equipment. In subsequent years, the numbers are positive because of the lower operating costs of the LAN. In this case, the payback occurs in 1993 after two years. The savings of $25,537 in 1992 and $27,651 in 1993 more than offset the initial cost of $37,360.

The discount rate used to compute the NPV in the example was 10%. If you compute a rate of return for projects (instead of, or in addition to, net present value), you must find the discount rate that exactly discounts the cash flows to make the cash inflows equal the cash outflows. In other words, solve for the interest rate that gives a net present value equal to $0.

The easiest way to do this is to let the computer program do the work. Look at Figure 6. Discounting the cash inflows and outflows in column H at 67% causes the net present value to be $9—very close to $0. This number was found by typing different numbers into the interest rate cell until the net presents value approached zero.

So, how can we use this number? Let's say your company borrows money from a bank to fund projects. If it costs 15% to borrow money, then you certainly want to earn at least a 15% return on the investment. Since this example gives a 67% return, the project would be accepted. This break-even characteristic makes the rate of return useful.

Figure 4.
Adjustment of After Tax Cash Flow

	A	B	C	D	E
35	Alternative #1: Install a LAN to run the application.				
36					
37		Total After Tax		Discounted at	
38		Cash Flow		10.0%	
39					
40	1991	$55,840		$55,840	
41	1992	797		725	
42	1993	(353)		(292)	
43	1994	1,101		828	
44	1995	1,254		856	
45	1996	2,383		1,480	
46					
47	**Totals:**	**$61,022**		**$59,436**	
48					
49	Net PV	$59,436			
50					
51					
52	Alternative #2: Continue to run the application on the mainframe.				
53					
54		Total After Tax		Discounted at	
55		Cash Flow		10.0%	
56					
57	1991	$18,480		$18,480	
58	1992	26,334		23,940	
59	1993	27,651		22,852	
60	1994	29,033		21,813	
61	1995	30,485		20,821	
62	1996	32,009		19,875	
63					
64	**Totals**	**$163,992**		**$127,782**	
65					
66	Net PV	$127,782			

This part of the worksheet adjusts the after tax cash flow for the time value of money.

Rate of return is not just an interest rate that expresses a return on the initial investment. The math involved in calculating the return also implies that cash inflows (or savings) are reinvested at the rate of return over the life of the analysis period. It is easy to understand that with a high rate of return this reinvestment assumption makes the numbers very questionable. This is not a problem with NPV. The reinvestment rate is the same for each proposal, and it is usually the cost of capital.

There is another reason why NPV is more appropriate than rate of return in this case. Most people would be more comfortable presenting the project by quoting the NPV numbers instead of the rate of return. The 67% rate of return is so high that it invites close inspection. As a rule of thumb, NPV is more conservative than rate of return.

Document Your Success

So, you used financial analysis to successfully sell the project. The project went in and was a huge success. Now, it's important to document the success and quantity the savings and benefits your company reaped. Especially with projects such as downsizing, it is highly unlikely that other people will beat the drum for you. Such projects involve so much organizational change that some people have unfavorable things to say even if the project was a complete success. These objections can be met with adequate post-cost documentation. There is no better way to mess up an opposing argument than with a few solid facts.

My company recently removed 100 minicomputers and replaced them with LANs—much like the LAN alternative in the example. The project was a success in every way. Management and end users had high praise for the LANs. It was a rare win-win situation. But, LANs changed the organization and changed the processing architecture of the company. Some people disliked this part of the project. Change frightens people; this is natural and understandable.

My company's accounting department did a detailed post-cost analysis. It quantified the success of the project and put hard numbers on what the project returned to the organization. We documented and tracked end-user productivity statistics for the year before the LAN and the first two years after. These numbers, combined with substantial intangibles from end-user comments, provided a powerful one-two punch that reaffirmed the project's benefits.

This type of analysis removes much of the emotion from decisions such as downsizing. Our particular project had a payback period of three months and a positive net present value of $13 million over five years. The savings came primarily from labor savings and secondarily from reduced hardware and software costs. Numbers like that speak volumes in supporting future projects and financial analysis credibility.

Figure 5.
Determining the Payback Period

	A	B	C	D	E	F	G
71		LAN		Mainframe		Difference	
72		Total After Tax		Total After Tax			
73		Cash Flow		Cash Flow			
74							
75	1991	$55,840		$18,480		($37,360)	
76	1992	797		26,334		25,537	
77	1993	(353)		27,651		28,004	←PAYBACK
78	1994	1,101		29,033		27,932	
79	1995	1,254		30,485		29,231	
80	1996	2,383		32,009		29,626	
81							
82	Totals	$61,022		$163,992		$102,970	

This part of the worksheet calculates the differential cash flow of the alternatives to determine the payback period.

Figure 6.
Calculating Rate of Return

	A	B	C	D	E	F	G	H
71		LAN		MAINFRAME		Difference		Difference
72		Total After Tax		Total After Tax				Discounted at
73		Cash Flow		Cash Flow				67.0%
74								
75	1991	$55,840		$18,480		($37,360)		($37,360)
76	1992	797		26,334		25,537		15,292
77	1993	(353)		27,651		28,004		10,041
78	1994	1,101		29,033		27,932		5,997
79	1995	1,254		30,485		29,231		3,758
80	1996	2,383		32,009		29,626		2,281
81								
82	**Totals**	**$61,022**		**$163,992**		**$102,970**		**$9**

This section of the worksheet allows a rate of return to be calculated by varying the interest rate through trial and error.

It's good idea to involve the accounting department in the original analysis. You probably will need them to handle depreciation, taxes, leases, and development of the cash flows. Since most accounting departments are responsible for doing the post-cost analyses of projects, it makes sense to involve them from the start. They will be much more charitable in auditing their own work.

With these tips in mind, you have enough information to begin the process of using financial analysis to support your next project. Even if the accounting people do most of the work, you can understand the process and help set the parameters used in the analysis. The results will make it much easier to sell your next big idea. I promise. ■

Soft Costs of LAN Implementation

Datapro Summary

The tangible costs of network implementation are usually obvious: the cost of the LAN components, including servers, workstations, and so on. But there are a number of hidden or "soft" costs that may not be obvious to network planners; training and support, primarily. If soft costs are ignored in the planning stages, the LAN will never be used to its full potential and will be a financial disappointment. By establishing a realistic list of soft costs, financial and productivity goals can be met.

It's interesting that the computer and communications industries, which are based on the difference between black and white (specifically 0 and 1), should have so many gray areas. One of the murkiest is determining a LAN's actual costs. In the last couple of years, network managers (and the users who generally foot the bill) have come to better understand and accept the hard capital costs of LANs. This is due to the tangible nature of most LAN components. With some reasonable planning, you can pretty well determine how many servers, cabling concentrators, workstations, application software packages, and network interface cards you will need to support "x" number of users.

The real challenge now is to better understand the less tangible costs involved in LAN implementation, training, and support. We must begin to get our arms around these soft costs so we can more accurately account for them. IS managers who can't account for soft costs find that their departments absorb these costs—and usually without the benefit of an adequate staff, since additional personnel cannot be justified. I have heard too many corporate IS managers complain that "LANs are turning into black holes right before our eyes, sucking up resources, people, money—anything that gets near them." By establishing a complete accounting of soft costs, users' expectations can be set appropriately, and IS can budget for the right level of support.

It's been my experience that soft costs are often completely ignored; it's as if they were a Pandora's box that many IS managers would rather not open. The main reason for this careful avoidance is that these costs have been poorly defined. I see them falling into five main areas.

The Cost of the Initial Business-Needs Analysis

A business-needs analysis should precede any hardware and software purchase, and will drive—and justify—the capital costs. For example, if the accounting department needs a real-time distributed database, you will probably need powerful CPUs and a higher-speed network. The time involved in analyzing the business need and mapping it to system requirements is a soft cost that must be accounted for.

The Cost of Sharing the Corporate Information Infrastructure

Unless a LAN is being installed in isolation, you will incur some costs in linking it to the enterprise network As a result, you need to calculate what percentage of the corporate

This Datapro report is a reprint of "The Soft Costs of LANs Can Hit Hard" by Mark Freund, pp. 27-30, from *LAN Technology*, Volume 7, Number 9, September 1991. Copyright © 1991 by M&T Publishing, Inc. Reprinted with permission.

infrastructure is required to support your LAN-based system. For example, the marketing department may need to access host-based data. Putting in a LAN-to-host gateway may impact host performance, place an added burden on a cluster controller or front-end processor, or require that you expand the backbone network. These costs need to be identified and quantified.

Tangential Costs, Such as Those Associated With Host-Based Software You May be Required to Purchase to Allow LAN-to-Host Data Exchange

Integrating heterogeneous workstations (which may be running different applications, operating systems, and protocols) with each other and/or with a host computer may mean you have to purchase specialized software to run at the workstation and/or host. Although this is largely a capital cost, it has a soft-cost component to it. For example, the host support staff will need time to install, test, and eventually upgrade the software on their system.

The Cost of Making the System Work

I call this "the cost of success." These are the costs incurred in resolving compatibility problems, adjusting your initial design, and so on. For example, you may have initially planned to put 50 users on a single token-ring LAN. However, reliability and performance requirements may dictate that you implement the system as two rings with 25 users each.

The Range of Support Costs

This includes ongoing maintenance, such as system backups, upgrades, and preventative maintenance. In this category, I also include the personnel costs for installing, testing, and optimizing the system, as well as the costs for training users.

Who's Going to Pay?

There are several dangers in not having a clear picture of soft LAN costs. First, since the costs are not accounted for, no one is asked to pay for them. Second, those who provide soft services, such as LAN support, wind up absorbing many of these costs without the proper resources, operating budget, and head count required to create and maintain a proper support infrastructure. Third, management and end users keep on buying LANs with inaccurate expectations as to what these systems actually cost.

I am in no way suggesting that these same LANs would not be bought if their soft costs were known, or questioning whether the business decisions to implement LANs are justifiable. I believe, though, that by not understanding the true scale of these complex technology acquisitions, many organizations are throwing their IS infrastructures into a potentially precarious imbalance, and doing little to temper increasingly inflated end-user expectations.

The primary problem in identifying soft costs is that there are no generally accepted guidelines or measurement tools to use as a reference point. Therefore, you will have to develop your own job-costing methodology for the purpose of measurement and accountability. You need to break down integration processes into sufficiently well-defined tasks to estimate their soft costs fairly closely. The methodology you establish will have to parallel your IS support infrastructure; more than likely, you will have to restructure IS to handle the new processes you define.

A Five-Step Solution

The following five steps can help you successfully account for soft costs. The cornerstone of this five-step approach is the job-costing methodology you create to break general LAN support issues down to their most basic elements, and then to assign costs to these elements.

Step 1: Dissect LAN Support to Establish True Costs

Currently, most organizations only quantify the hard capital costs of hardware and software acquisition. The few companies that do quantify the soft support costs tend to use a relatively simple amortizing approach, where they divide the cost of end-user computing support by the number of users in the organization and then multiply the result by the number of users in a specific department. This number is used to reflect a specific department's share of the overall LAN support cost.

Unfortunately, this type of accounting does not reflect the actual support costs; it is unfair as well as uninformative. For one thing, it doesn't account for the solution level being provided. For example, there may be only 10 users in marketing who need to use Macintoshes, while all other users in the organization use DOS-based personal computers. The cost per user of integrating the Macintoshes with the host is considerably higher than the cost per user of providing the DOS systems with host access.

However, if the costs of providing host access are lumped together and simply divided by the total number of users, the DOS users end up paying a higher cost than they actually incurred. By using the simple division method of allocating costs, the effectiveness and the value of the support being provided by the current support group becomes difficult or even impossible to measure.

You need to dissect or break up soft costs if you hope to allocate them appropriately. Specifically, set up each of the services offered by the support group as a specific product and have end users fill out a request for services to be performed when support is needed. Map each request against your support staff's forecasted and actual costs; this will allow you to build a database of empirical information that has a sufficient degree of granularity that you can then construct a useful support cost model. Even if this model is used for accounting purposes only, the information it provides will prove invaluable.

Step 2: Group Individual Support Costs to Establish Guidelines

Next, you should define a consistent set of guidelines for cost measurement. The goal is to assemble all the associated component costs into one logical functional cost. For example, group all costs associated with attaching a Macintosh to a host.

This set of valuation rules will be built out of different sets and subsets of the services identified in Step 1. Once you've identified specific costs, grouped them, and defined valuation rules, you can build a template that users can employ to forecast the potential soft costs (both start-up and operational) when budgeting, and IS can use to ensure that it is funded properly to support the defined environment.

Step 3: Attribute the Individual and Grouped Support Costs to the Functions and People That Bear Them

End users have individual desires and business requirements. Therefore, it is imperative that they have flexibility when it comes to making LAN platform and applications decisions. For example, some users may want (and can justify for business reasons) Macintoshes as their LAN workstations even though 85% of the organization uses IBM PC compatibles.

It is important, however, that both the end user and IS recognize that the cost of swapping a Macintosh for an IBM PC clone is not limited to the price difference (if any) between the two machines. The costs of providing equal functionality between dissimilar environments and ongoing support must be recognized up front, so IS will be prepared to provide the appropriate support and the end user will be prepared to pay for it. The bottom line is that the real costs of support must be attributed to the systems and people requiring it.

Step 4: Inflate the Actual Support Cost Estimates the First Time Around

Be reasonable in estimating support costs, but a 25% to 50% contingency factor is not unrealistic the first time around. This is primarily due to the unknowns that are characteristic of any new job-costing system. Once you have real cost numbers, start plugging them in. Overages to the original forecasts should be accounted for on an instance-by-instance basis. This will provide valuable empirical evidence for the next budgeting period and for the evaluation and adjustment of future support cost forecasting.

Step 5: Re-Establish the Cost Picture for LAN Support

This final step is the most important. You must use this cost modeling as an opportunity to get end users' expectations for LAN performance in line with actual costs. There will certainly be some resistance at first—no one likes to be responsible for costs. However, the benefits of being able to build a more tailored and responsive LAN support infrastructure greatly outweigh these concerns.

One warning, though: IS must deliver. It must apply the resources promised and budgeted. To help ensure success, be sure to communicate with your end-user community (and their management as well as yours) each step of the way.

An Increased Return-on-Investment

Cost accountability has been a major problem within IS departments historically. This has been primarily due to the lack of communication between the IS group and users they have served. In addition, the centralized nature of a host computer has complicated the process of calculating the costs of the departments and individuals who share its applications and resources.

The move toward distributed systems should make it easier to identify and allocate soft costs. Since users tend to be behind the acquisition of personal computers and LANs and have to justify their costs, it is easier to align the costs and benefits of these systems with the users who purchased them. But first, we must recognize their real costs (both hard and soft) so that when end users make their systems decisions, their expectations are realistic and the proper support infrastructure is in place to support these decisions. The challenge will be to preserve a high degree of cost granularity as IS moves toward a more centralized LAN infrastructure.

Getting a handle on soft costs means that a more accurate "I," or investment, will provide the basis for a more credible "R," or return, in management's return-on-investment (ROI) equation. This will help ensure more realistic expectations, better justifications for IS investment, and increased order and balance throughout the company.

So if some users don't like black or white, but prefer pink, they can have it—they just need to pay for it. ■

Manager's Guide to Cabling

In this report:

Datapro Summary

Selecting cabling for local intrabuilding installation can be one of the most traumatic experiences for a telecommunications manager. In the past, the decision was relatively simple: All communications involving terminals and computers took place over standard voice grade telephone lines. Today's environment, however, is significantly more complex. Facing requirements for local area networks and emerging technologies and standards, telecommunications managers must treat cabling as a strategic resource and be aware of its inherent capabilities as well as its limitations. This report discusses specific media alternatives; reviews the new EIA/TIA wiring standard and alternative proprietary implementations; and provides guidelines for specifying, installing, and managing intrabuilding cable plant.

Effective decisions on cabling a facility must be predicated upon the physical construction, electrical characteristics, advantages and disadvantages, installation issues, and costs of different media types. Recent cabling events affecting installation decisions include the approval of the EIA/TIA-568 Commercial Building Telecommunications Wiring Standard; the advent of unshielded twisted-pair Ethernets; the development of the 16M bps token-passing ring; the standardization of LAN bridge technology; the maturation of optical fiber implementations; and the development of "wireless" LAN alternatives.

Introduction to Cabling

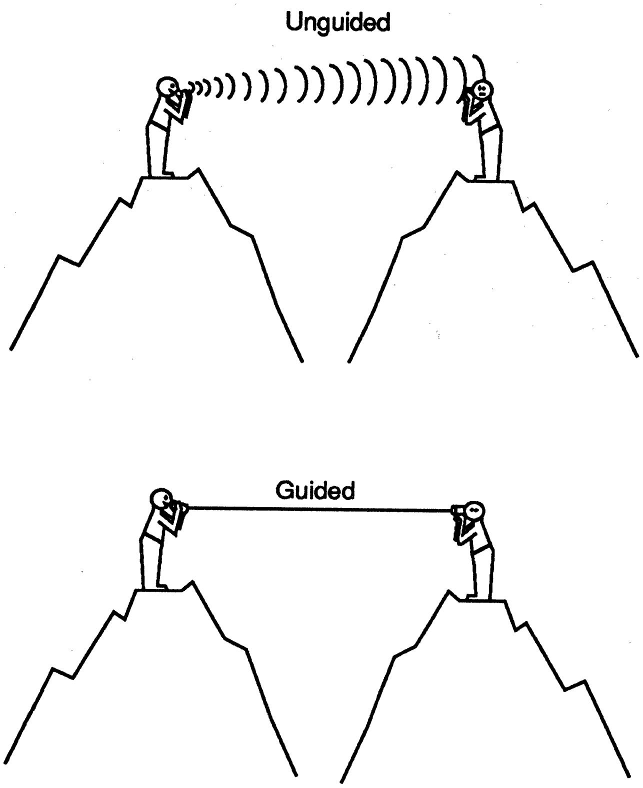
Guided versus Unguided Media

The first decision one usually makes regarding media is whether it is to be "guided" or "unguided." All physical transmission media fall into one of these two classes. Those signals that radiate through the air, radio or line-of-sight optical transmission, are generally considered to employ unguided media. On the other hand, different forms of wire—whether they be copper, aluminum, steel, or even optical fiber, and whether they be paired wires or coaxial cables—are all guided media. Figure 1 illustrates the differences between guided and unguided media. Discussion of *on-premises cabling* is normally limited to guided media, although there are some leading-edge, state-of-the-art implementations that employ unguided media indoors. One should be aware that there are those who believe that all media, including airwaves and line of sight, are guided, but that the waveguides in these cases are simply less stable than wire.

Any digital data or analog voice information can be transmitted over single or paired wires. Although once commonly used in the world of telephony, single-wire systems using the earth as a return conductor have fallen into disuse. Two paired wires employing balanced electrical transmission are generally more reliable, since they are less vulnerable to induced electrical noise. While coaxial cable is not truly a balanced medium, it does provide some additional benefits, since the outer shield reduces the emission and absorption of radio frequency and electromagnetic interference

—By Michael L. Rothberg
President
Applied Network Solutions, Inc.

Figure 1.
Guided versus Unguided Media



(RFI/EMI). This report discusses each of these media in more depth in later sections.

The term *waveguide* usually connotes copper or metallic media. Optical fibers, in fact, are true waveguides in every sense of the word, possibly even more so since they neither emit nor absorb any form of electrical signals.

Plenum versus Nonplenum Cabling

Focusing on guided media, one should recognize that there are basically two methods of installing these wires within a building. Wire can be installed in airspace above the ceiling—the plenum—or it can be installed in conduit or underfloor troughs. One would normally opt for the plenum space to avoid the additional cost of conduit and the limited space of floor troughs. Simple enough? The National Electrical Code (NEC), however, states that all cable installed in plenum space must be installed in conduit, unless it has been certified by an approved agency as having fire-resistant and low-smoke-producing characteristics.

The next question is how to achieve this level of fire rating with insulating or dielectric material. Two forms of dielectric are commonly used today: the first is polyvinyl chloride (PVC), which is not classified as "low-smoke low-flame" and the second is a fluoropolymer (such as Teflon—a registered trademark of Du Pont), which is classified by the Underwriters Laboratories as fire resistant. Other cables having similar dielectric materials may also be classified as "low-smoke low-flame" (such as HALAR by Ausimont, KYNAR by Pennwalt, TEFZEL by Du Pont). One significant advantage of Teflon-coated cables is that they provide superior electrical performance over other plenum and nonplenum cables. In addition to performance, in many cases, plenum cables offer significant cost advantages over nonplenum cables, discussed next.

Cost—Materials/Labor

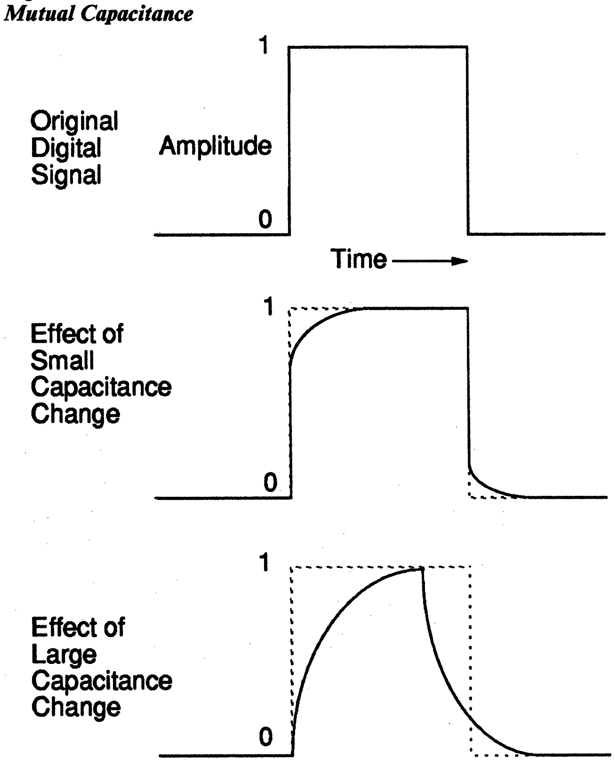
One can quickly establish that the actual materials cost of plenum cable is much greater than the cost of nonplenum cable, sometimes on an order of 3.5 to 1. It is not unreasonable to encounter differences of as much as 350%. For example, PVC-coated coaxial cable may cost approximately $0.22 per linear foot compared to $0.98 per linear foot for comparable plenum cable. These costs are deceiving, however, since the cost of the actual cable comprises a small part of the entire installation job.

It is likely that when the costs of conduit, conduit connectors, couplings, and straps are added to a job that the cost of nonplenum cable will be much greater than its plenum counterpart. For example, 1 foot of conduit (purchased in quantity of 600 feet) may cost $2.05. Without considering the cost of the other components, the nonplenum cable has risen to over twice that of the plenum version.

By reviewing a job estimate, one may discover the following:

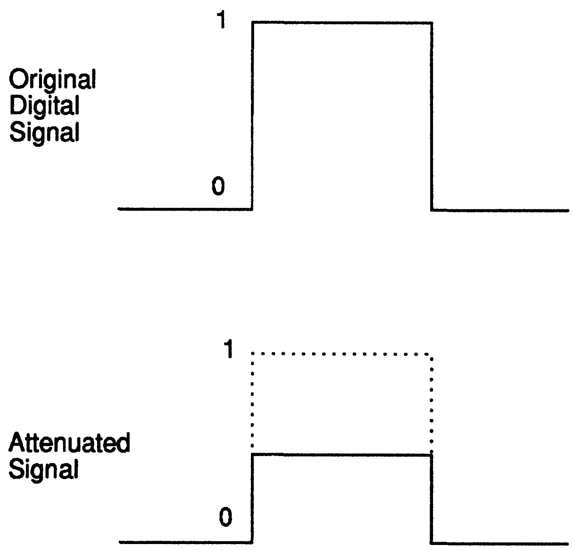
1. There are more fittings, adapters, and components required to install nonplenum cable in conduit than plenum cable without conduit.
2. Significant increases in labor cost are inherent in any installation requiring conduit.
3. Plenum cable systems can be installed more rapidly, resulting in more timely service to users.

Figure 2.
Mutual Capacitance



Effect of mutual capacitance on a digital signal.

Figure 3.
Attenuation



Effect of attenuation on a digital signal.

Performance

In the modern office environment, telephone lines and other media are often used to transmit digital data as well as analog voice information. These mixed transmission requirements frequently require longer cable runs, which in turn result in requirements for superior electrical performance. Low signal distortion; minimal interference from adjacent wiring; and clear, error-free transmission are necessary for effective transmission of digital data.

The dielectric insulating material will directly affect the cable's basic performance. The three electrical properties of cable that are affected by the dielectric, and will in turn determine the quality of the signal, are the following:

1. Capacitance
2. Attenuation
3. Crosstalk

Capacitance is a measure of the cable's capability to store an electrical charge and to resist sudden changes in the magnitude or voltage of that charge. Capacitance is a function of the thickness and the dielectric constant of the insulation material. Capacitance exists between different pairs of wires, between two wires of the same pair (mutual capacitance), and even between the inner and outer conductors of coaxial cable. Figure 2 illustrates the effect of mutual capacitance on a digital signal.

Attenuation is a measure of the reduction in signal strength, or amplitude, as the signal propagates down the medium. It has a direct impact upon the capability of the receiver to distinguish between one and zero bits or signal elements, since abrupt changes in amplitude are the basic criteria for determining the bit or signal element value. Figure 3 illustrates the effect of attenuation on a digital signal.

Figure 4 depicts the combined effect of both capacitance and attenuation on a digital signal. Repeaters are used to retransmit the signal and restore it to its original values. For this to be effective, the repeaters must be placed close enough to each other on the line; otherwise, not enough information would be available in the remaining signal to determine that it was in fact a bit. Thus, the use of plenum cable may obviate the frequent placement of repeaters.

Crosstalk is a measure of the signal that is absorbed by a "quiet" pair of wires when adjacent to or in the vicinity of an "excited" pair. Plenum cables with a lower dielectric constant have less capacitance unbalance and result in lower crosstalk than nonplenum cables.

Specific Cable Types

Overview

This report provides an overview of cable types commonly used in intrabuilding wiring schemes. While not exhaustive, due to the wide variety of different cable types and sizes, the characteristics of three primary cable types are explored:

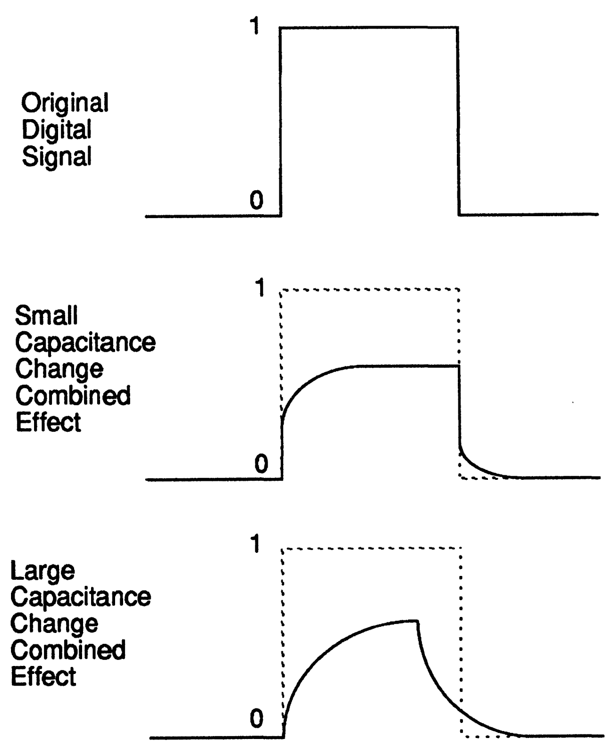
1. Twisted pair (both unshielded and shielded)
2. Coaxial cable (broadband and baseband applications)
3. Optical fibers (single mode and multimode)

Analysis Criteria

For each of the preceding types of cable we explore the following issues:

1. *Physical Construction*—The medium's physical construction includes those characteristics such as the number of conductors, the conductors' composition, the type of insulation, and any other pertinent distinguishing characteristics.

Figure 4.
Mutual Capacitance and Attenuation



Combined effect of mutual capacitance and attenuation on a digital signal.

2. *Applicable Transmission Technologies*—Transmission technologies include a discussion of whether the medium can support digital and/or analog transmission, the type of modulation generally employed, and, where applicable, the frequencies at which signals are transmitted. Baseband transmission employs digital signaling and encoding techniques (such as Manchester and Differential Manchester Coding), while analog transmission employs Frequency Shift Keying (FSK), Phase Shift Keying (PSK), and Amplitude Modulation. Another technique is used to digitally code analog information (such as voice). This process, known as Pulse Code Modulation (PCM), is actually a method of sampling and encoding information. It is not an analog modulation scheme as the name might suggest.
3. *Geographic Scope*—The medium's geographic scope describes the maximum feasible distance between two nodes. These limitations cannot be discussed alone, as geographic scope is also related to bandwidth and transmission technology.
4. *Bandwidth*—The essence of bandwidth is the information-carrying capacity of the channel or medium. Wherever possible, one should define bandwidth in terms of useful information (e.g., bits per second). Bandwidth can more technically be defined as the width of the signal spectrum or the difference between the limiting frequencies of the channel. This is frequently expressed in hertz (Hz), which is a measure of the number of analog cycles per second. The terms "Hz" and "baud" are meaningless in describing a channel's bandwidth, as they are not directly equivalent to bits per second.
5. *Connectivity*—This discussion on connectivity focuses on whether the medium is generally used for point-to-point or point-to-multipoint (broadcast or multiaccess) transmission. Due to specific physical characteristics and implementation constraints, some transmission media lend themselves to either broadcast or point-to-point connectivity or both. We will highlight those attributes of the media that affect these capabilities.
6. *Noise Immunity*—There are several channel parameters that influence the integrity of the information it is carrying. Distortion, interference, and noise are all network contaminations that alter the signal shape. All are unintentional phenomena that must be considered by network designers. Distortion is any modification of the transmitted signal that is caused by varying effects of the transmission system on the signal itself (e.g., capacitance, attenuation). Interference is contamination resulting from external energy that affects the signal's integrity. Noise encompasses either or both of these internal and external disturbances; it in turn causes abnormal processing and reception of the signal. Most noise results in very small variations of the signal, on the order of microvolts. If the impact on the signal is greater than this, then corrective action must be taken. The signal-to-noise ratio of a specific medium is a convenient measure of the circuit's reliability.
7. *Security*—Depending on the application requirements, different levels of security are built into a network (such as access to the facilities, access to the network, and access to data). Our concern is with unauthorized access to the network medium by tapping. Some media are very difficult to tap, whereas others almost invite intrusion. We will discuss each medium in terms of ease of tapping and also whether or not signals are emitted that can be intercepted by intruders.
8. *Installation Considerations*—A number of issues surround installation. We evaluate each medium in terms of its suitability for intra-building wiring, focusing on such characteristics as ease of installation, tapping and splicing, and connectors.
9. *Applications*—Some media lend themselves to certain types of networks (such as factory floor versus office environments). The associated issues are highlighted, enabling readers to develop a view of the most effective applications.
10. *Cost*—Cost is probably the most difficult element to evaluate. Bear in mind that the cable itself is a small part of the total cost. The labor costs associated with installation, as well as ongoing management and maintenance costs, will usually dwarf cable cost. In addition, the range of installation costs varies so widely for different implementations and sites that it is almost impossible to establish any benchmarks. Recognizing that there may be significant variances, we attempt to identify the causal factors while establishing some midrange costs for those supposedly "normal" installations.

Specific Medium Analysis

Twisted Pair/Shielded Twisted Pair

Physical Construction—The most pervasive communications medium is the twisted pair. The wires themselves are twisted together in a regular geometric patter, thus making the electrical properties constant; this also reduces the noise and adjacent interference for the entire length of the line. Twisted pairs can also be purchased with braided or foil electrostatic shielding, which further enhances the noise reduction properties of the medium.

Most commonly used unshielded twisted pairs have a conductor thickness of American Wire Gauge (AWG) 24 or 26. These are often used for telephony connections. The characteristic impedance of these pairs is typically 100 ohms. The characteristic impedance of a line defines the relationship between voltage and current at any point along the line and must be matched to the characteristic impedance of the connectors and attached equipment. A mismatch in the impedance of these components will result in signal reflections on the wire.

Shielded twisted pairs are often AWG 22 and have a characteristic impedance of 150 ohms. The IBM cabling system is representative of implementations using this type of twisted pair. While IBM is a proponent of shielded twisted pair, other vendors adopt the position that unshielded wire is more effective. We explore this issue in context when discussing the AT&T SYSTIMAX Premises Distribution System.

Applicable Transmission Technologies—Twisted pairs support both analog and digital transmission using a wide variety of encoding and modulation techniques. Digital transmission on twisted pairs often employs PCM when employed for integrated voice and data applications.

Geographic Scope—Twisted-pair installations are frequently limited to one to two kilometers without repeaters.

Using repeaters to compensate for attenuation, the line can be extended to meet most practical application requirements.

Bandwidth—As with many other media, the bandwidth and distance of twisted pairs are interrelated. Other factors affecting the bandwidth capabilities of the twisted pair are the signaling techniques used, the quality of the dielectric material, and the precision of the manufacturing process. The use of repeaters at relatively short intervals in a LAN environment can enhance the maximum bandwidth and enable transmission rates on the order of 10M to 16M bps. Experimental implementations exist over very short distances at signaling rates of 100M bps (using the Fiber Distributed Data Interface—FDDI—protocol). A practical maximum used in digital PBX implementations is 1.544M bps. Synchronous user terminals operate between 9600 and 19,200 bps, while asynchronous devices usually operate below 9600 bps.

Connectivity—Twisted pairs are generally employed in point-to-point environments, but it is possible to use them in configurations that simulate multipoint broadcast capabilities. This is achieved using a technique known as "daisychaining," where each interface device receives an incoming pair on one port and has an outgoing pair on another port. The device is now responsible for forwarding the information. In a true broadcast environment, the device only copies information that is destined to itself. It does not interfere with the continuing propagation of the signal down the medium.

Noise Immunity—Twisted pairs provide the least immunity to noise of all physical conductors. The medium both emits and absorbs noise, and it is sensitive to both Electro-Magnetic Interference (EMI) and Radio Frequency Interference (RFI). Reasonably good noise immunity can be achieved if the twist length is significantly shorter than the effective wavelength. In balanced systems operating at low frequencies, noise immunity can be as effective as more reliable media, such as coaxial cable, but above 100kHz coaxial cable will usually provide superior performance.

There is significant controversy surrounding the issue of shielding for protection against EMI/RFI. This issue will be explored in the discussion of proprietary vendor architectures.

Security—Because twisted pairs emit signals, they can be tapped easily. In addition, their pervasiveness and accessibility further increase their vulnerability. Because of this, they are considered to be the least secure of the guided media. Techniques such as encryption, however, can add value at the higher level of data protection as opposed to access to the network.

Installation Considerations—Twisted pairs will usually be easier to install than heavier coaxial cable. In addition, it may be possible to use previously installed wiring, since a considerable number of lines lie "dead" under the floor in many office buildings. As with most media, these lines are not removed when devices are relocated. The medium's electrical characteristics will be an important determining factor in whether these circuits—previously used for low-bandwidth analog voice—can be used for high-speed digital signaling (i.e., LANs). It is essential that manufacturers' specifications for twisted-pair LANs be scrupulously observed and that the wiring conforms to new standards such as the EIA/TIA-568 specification (discussed in a later section).

Conversely, there may be excess wiring in the floor, crowding the troughs and inhibiting the installation of new circuits. These issues are obviously implementation dependent, but bear in mind that some LAN implementations facilitate the replacement of hundreds or even thousands of twisted pairs with one broadcast coaxial cable. Effective management of these resources is discussed in the section Cable Plant Management.

Applications—There are numerous applications for twisted pairs. PBX and conventional telephone systems are by far the most common. Although the dominant application is telephony, twisted pairs are frequently used for point-to-point terminal connections and can be effectively used in some LANs.

Coaxial Cable

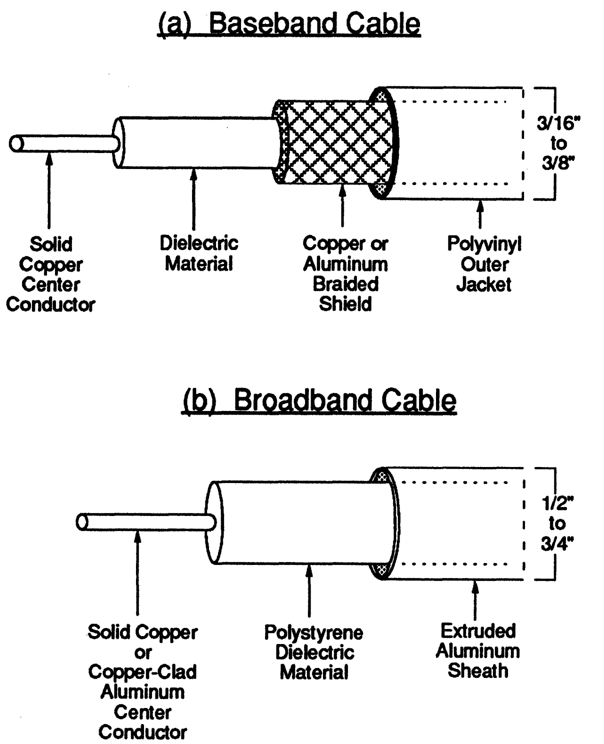
Physical Construction—Coaxial cable has been the dominant transmission medium for LANs for many years, although it is rapidly yielding its entrenched position to unshielded twisted pair. The cable is composed of two conductors in concentric circles to each other. A solid wire forms the inner conductor, while the outer conductor serves as a shield and is usually grounded. Between the two conductors is a dielectric material. There are many different forms of coaxial cable; however, they are usually divided into two classifications—those used for baseband transmission, and those used for broadband transmission. Keep in mind that the terms *baseband* and *broadband* denote the transmission technology but are sometimes used to ambiguously define the different cable types—a tradition which it is more difficult to depart from than to maintain.

The primary distinction between the two cable types is the characteristic impedance. Baseband cable has a characteristic impedance of 50 ohms, while broadband cable is 75 ohms. Figure 5 illustrates the construction of broadband and baseband coaxial cables. Baseband cable employs a center conductor of solid copper. This is surrounded by a dielectric material, which in turn is contained in a braided aluminum or copper shield or outer conductor. The entire package is then encased in an outer cover made of PVC or a fluoropolymer. The diameter of the cable varies with different implementations but may range from 3/16 to ⅜ of an inch.

Broadband cable, on the other hand, is essentially the same 75-ohm cable that is used in Community Antenna Television (CATV) Systems. The inner conductor is often composed of copper-clad aluminum (as opposed to solid copper) as a cost consideration, since the cable run may be very long. Copper-clad aluminum is satisfactory as a conductor, even though it does not have the same properties as solid copper. The reason for this is a phenomenon known as "skin effect," where the current distribution or the signal tends to migrate to the outer surface of the conductor as the frequency rises, thus obviating the requirement for the conductive properties of solid copper. The center conductor is then encased in a dielectric polystyrene material. The whole assembly is then covered with an outer conductor of extruded aluminum. Outer insulation is usually not required on broadband cables installed indoors. The entire cable may be either ½-inch or ¾-inch thick depending, upon the nature of the cable run (trunk, distribution, or drop).

Applicable Transmission Technologies—As with the physical cable types, there are also distinctions between baseband and broadband transmission systems. In baseband systems there is no modulation of the signal. Transceivers are used to place digital signals directly onto the

Figure 5.
Coaxial Cable



Physical construction: (a) Baseband coaxial cable (b) Broadband coaxial cable.

cable. These signals are encoded using some of the techniques identified earlier, usually by means of Manchester or Differential Manchester phase encoding. The digital signal occupies the entire bandwidth of the cable; thus, baseband cables have only one channel in operation at any moment in time.

In broadband systems, analog signals are transmitted on the cable using both frequency- and phase-modulation techniques. Multiple frequency-derived channels are operated on a single cable simultaneously. The cable's frequency spectrum is usually divided into a forward and return spectrum. Figure 6 illustrates a single-cable midsplit system and a dual-cable system.

The cable can be viewed as a two-way radio frequency system. A device connected to the cable can transmit a signal in one direction on the return spectrum and receive the same signal as it is retransmitted from the headend on the forward spectrum. The headend functions as a retransmission facility and steps up received signals in the return spectrum to the frequency of the corresponding channel in the forward spectrum. Figure 7 defines the relationship between the broadband cable plant components.

Geographic Scope—Broadband systems can cover distances of up to 40 kilometers as a practical limit; however, note that the IEEE 802.3 local area network standards specify a maximum distance of 1,800 meters from the headend. By placing the headend at a central point, end-to-end cable distances of 3,600 meters can be supported.

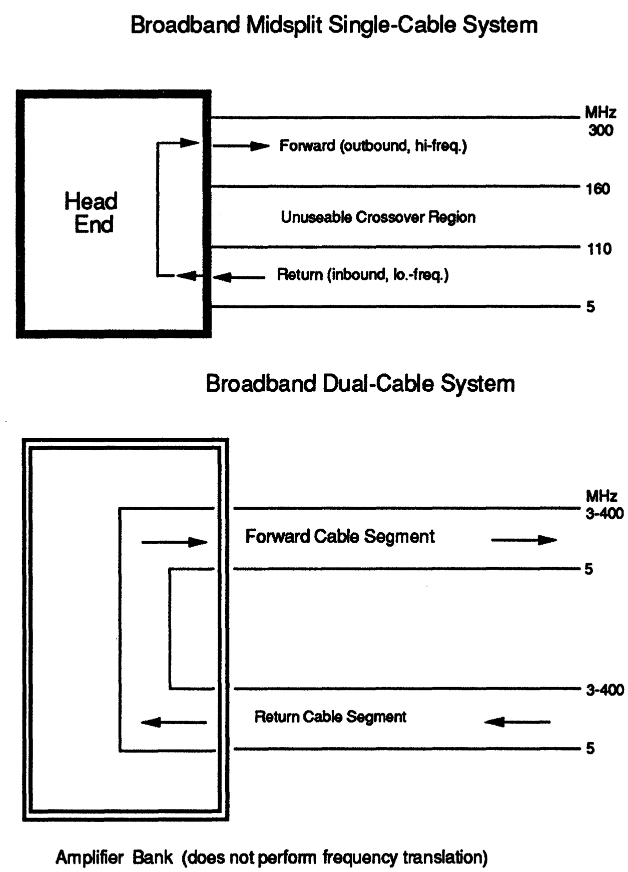
Typical baseband systems can cover distances of between one and three kilometers. The 802.3 limit for a baseband network is 2.8 kilometers between any two communicating devices. The actual cable segments may total more than this, as illustrated in Figure 8. The primary reasons for these limitations are not based upon the physical cable construction, but rather the timing considerations in the channel access protocol.

Bandwidth—Bandwidth is another variable that differs significantly between broadband and baseband systems. Most baseband implementations operate in the range of 10M bps, recognizing of course that there are exceptions (such as token-ring at 16M bps, Network Systems Corp. Hyperchannel at 50M bps). Broadband systems, on the other hand, offer significantly increased bandwidth capabilities due to the fact that there are multiple channels on each cable. Typical broadband implementations with highly saturated cables may yield aggregate bandwidths as high as 100M pbs.

A typical 300MHz midsplit system can support 110- to 120MHz in each direction, but one must allow for guard bands to protect the system from such phenomena as adjacent channel splatter (the "slopping over" of signals from one frequency-derived channel to another). In addition, an unusable crossover region of approximately 50MHz is required. Where video requirements exist, 6MHz must be allocated for each video channel. These requirements can rapidly diminish what initially appears to be "unlimited bandwidth."

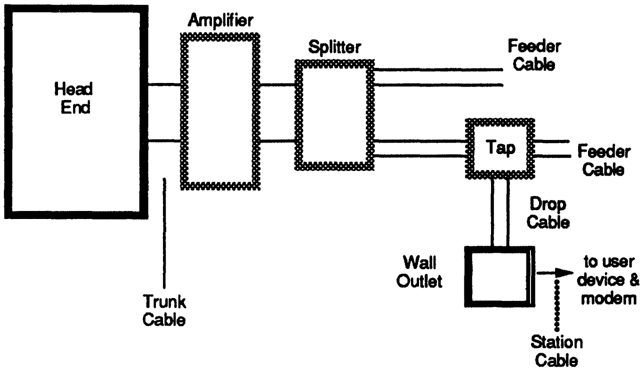
Connectivity—Coaxial cable can be used in both a point-to-point mode and a multipoint mode. In local network bus architectures, multiple devices are usually

Figure 6.
Broadband Spectrum



Broadband frequency spectrum: midsplit and dual-cable systems.

Figure 7.
Broadband Components



Broadband system component relationship.

"dropped" from a single cable. Depending on the application and the required data rates, a broadband system may support thousands of connections. The system must be balanced, however, to compensate for the loss of signal strength at each tap (insertion loss) as well as normal attenuation. Amplifiers are usually placed at intervals of 0.5 to 1.5 kilometers to regenerate the signals.

Baseband systems can support up to 100 devices for a single cable segmenting some implementations (such as the IEEE 802.3 10BASE5 "thick" Ethernet equivalent). As with broadband systems, one must also be concerned with the issue of insertion loss. Proper repeater placement is essential to signal regeneration.

Noise Immunity—Coaxial cable provides for superior noise immunity over conventional twisted pairs. As with other media, its immunity to noise is subject to the impact of variables such as the application and the environment. Baseband systems usually provide immunity of 50 to 60 decibels (dB), while broadband systems operate with 85 to 100 dB.

Security—Both broadband and baseband cables are subject to some of the same security issues as are twisted pairs, but to a lesser degree. Since the cable does not emit as high an amplitude signal, tapping or intercepting the signal will be more difficult.

Installation Considerations—Coaxial cable is significantly more difficult to install than twisted pairs, but the potential costs associated with the installation may be offset by the multidrop nature of the cable. Other considerations relating specifically to broadband implementations include the difficulty in handling (caution to avoid crimping of the outer extruded aluminum sheath) and the fact that amplifiers, taps, and splitters may not fit easily in the floor raceways of office buildings. One should carefully measure the dimensions of the floor raceways and plenum space before assuming that an installation is feasible. In addition, cables of different physical construction characteristics are usually used for trunk, feeder and drop, and station cable applications.

Applications—Coaxial cable is useful for office automation applications where many devices may be multidropped from one cable. Other applications include both factory and laboratory automation. In fact, one can use broadband coaxial cable to support multiple networks in a factory environment, thus separating administrative and process control networks on different frequency-derived channels while still residing on the same physical cable.

Coaxial cable's flexibility in terms of its high bandwidth potential makes it an attractive intrabuilding wiring alternative for backbone applications. These applications usually include riser (vertical) cables as well as horizontal cables connecting wiring closets and equipment rooms. It is becoming less popular for connecting devices in the work area primarily because of the increased performance attributes of high-quality twisted pair. It is interesting to note, however, that thin coaxial cable (RG-58) used in "thin" Ethernet applications may in fact be less expensive than twisted pair in some scenarios.

Cost—The installed cost of coaxial cable exceeds that of conventional twisted pairs. Baseband cable is usually less expensive that broadband cable, with the exception of some specific types of multiple-shield baseband cable (e.g., thick Ethernet). In either case, the more significant cost component is installation. It is reasonable to assume that the installed cost of these cables will usually be less than $4 per linear foot; however, some quotations may dramatically exceed this figure. One should be aware of building code restrictions that may adversely affect labor costs. In addition to the cost of installing conduit, modifications to floor troughs and drywall partitions for taps and splitters, as well as drilling through many floors for riser cables, will all contribute to the total installation costs.

Optical Fiber

Optical fiber technology has taken several great leaps forward in the past few years. The advent of standard implementations such as the Fiber Distributed Data Interface (FDDI), a 100M bps token-ring, along with its proliferation as an essential element of backbone wiring plans, has catapulted the technology to a prominent position as a candidate for many implementations. Not to be ignored are the dramatic decreases in the cost of connectors and easier connectorization methods. The adoption of fiber as an element of the new EIA/TIA-568 specification has further catalyzed the "fiber fervor."

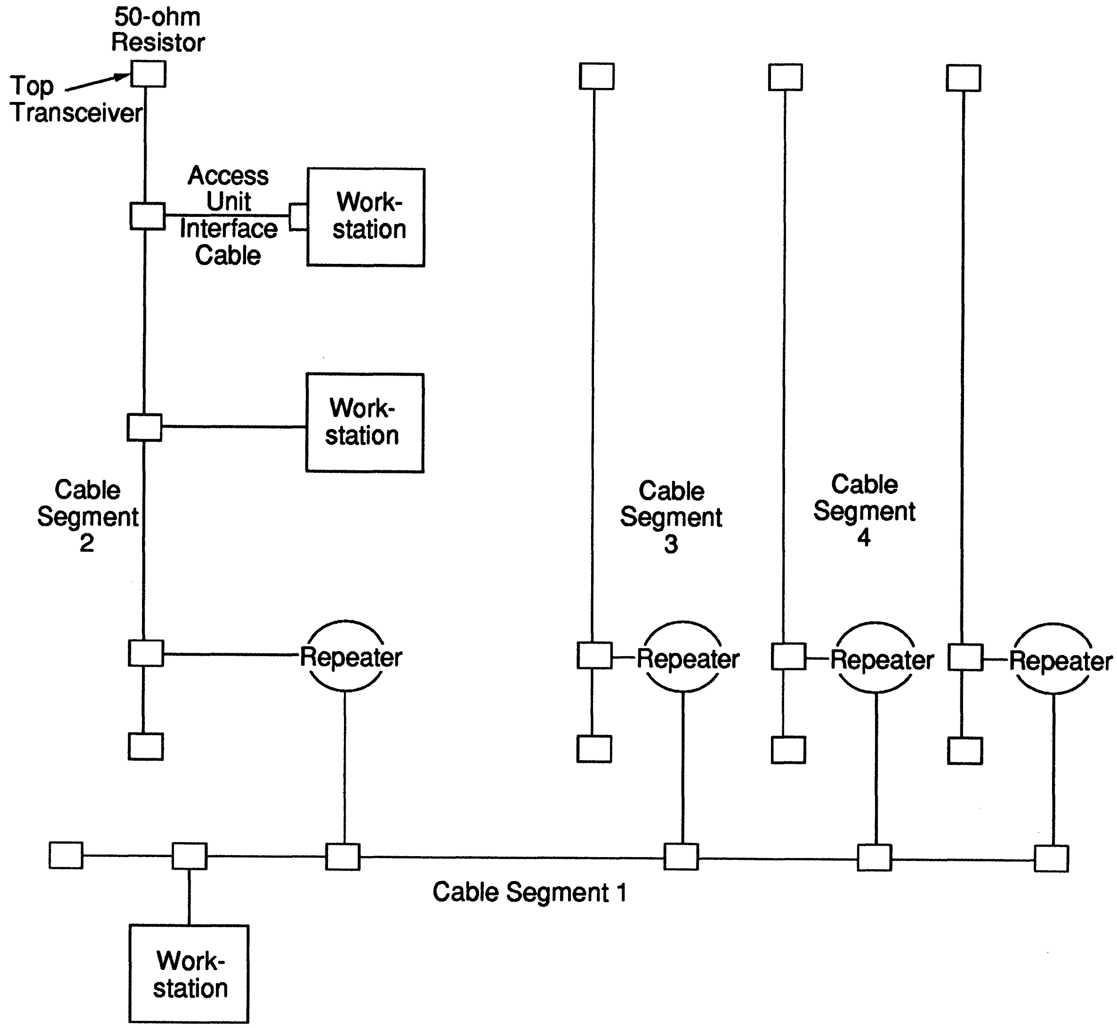
Physical Construction—Transmitting signals on optical fiber is accomplished by sending a signal-encoded beam of light through a glass or plastic fiber or waveguide. Each 125-micron-thick fiber has a center core of highly refractive glass or plastic. Surrounding this core is a cladding material with a somewhat lower index of refraction. The goal is to contain as much light as possible within the center core so as to facilitate total internal reflection. The cladding serves to isolate adjacent fibers from one another and thereby prevent crosstalk. Covering the fiber is an absorptive material that forms the outer surface of the fiber. Figure 9 illustrates the construction of an optical fiber.

Several different types of optical fibers are available:

1. Single-mode step index fiber is characterized by an extremely small core diameter on the order of 8 microns.
2. Multimode step index fiber is characterized by a wider diameter core permitting multiple propagation modes.
3. Multimode graded index fiber differs in that the index of refraction falls off in gradual steps from the center of the core to the outer surface.

The effect of the different core diameters of single-mode and multimode fibers is significant. In multimode fibers, the diameter of the core, usually 62.5 microns (other implementations of 50, 85 and 100 microns exist), is sufficiently large to allow light to travel on different paths. Each

Figure 8.
Baseband Bus



Multisegment baseband bus architecture.

of these paths is considered a mode of propagation. The photons that constitute the light pulse all have the same operating wavelengths, but depending on where the first reflection occurs, they take different paths through the waveguide.

Single-mode fiber, on the other hand, is characterized by a core diameter of approximately 8 microns. The salient point here is that the core diameter is nearly the same as the wavelength of the light. Thus, the photons are only permitted one route through the waveguide. Figure 10 illustrates these different types of fiber.

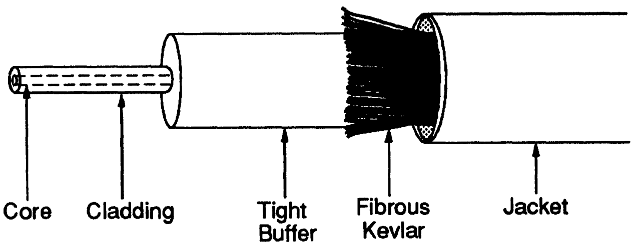
Optical fiber cables may contain multiple fibers. Some cables have steel or composition stress members among the glass fibers to provide support when the cables are suspended overhead. Other cables contain fibrous material (Kevlar) of very high pulling strength, the purpose of which is to permit pulling the cable without stressing the optical fibers.

Applicable Transmission Technologies—Both semiconductor lasers (injection-laser diodes, or ILDs) and light-emitting diodes (LEDs) are used as light sources for transmission. LEDs are predominantly used in data transmission environments on multimode fiber, while lasers are ideal for voice transmission on single-mode fiber. ILDs produce a very narrow beam of superradiant light and provide more bandwidth over greater distances. LEDs are less expensive, not as sensitive to temperature, and boast a longer operational life.

The detectors on the other end of the fiber function by converting the light pulses into an electrical signal. Two photodiode devices are used. The first is a PIN (used in conjunction with LEDs) photodiode where an intrinsic (I) region is sandwiched between the P- and N-doped semiconducting regions. The second is the Avalanche photodiode (APD—used in conjunction with ILDs), which uses a stronger electric field. For the most part, the PIN detector is less expensive than the APD, but at the cost of not being as sensitive.

Transmission on an optical fiber is usually unidirectional, although recent developments enable transmission of multiple signals on a single fiber in either direction simultaneously. LANs generally require the use of two fibers. The development of Wavelength Division Multiplexing (WDM) has progressed to the point where up to 10

Figure 9.
Optical Fiber



Basic optical fiber construction.

separate channels of information can be transmitted simultaneously on one fiber. A WDM device is constructed of a number of optical filters that can extract several discrete wavelengths from a single fiber. Practical implementations are generally limited to two channels.

Geographic Scope and Bandwidth—With optical fibers, it is virtually impossible to discuss either geographic scope or bandwidth separately. Several variables affect the range of optical fibers:

1. Bandwidth
2. Repeater placement
3. Fiber type (single or multimode)

Multimode fibers without repeaters can be extended to approximately 10 kilometers. In this range, the fiber can support transmission rates of up to 100M bps. For shorter distances of perhaps one kilometer, up to 1G bps has been demonstrated.

Single-mode fiber, although not used very often in short-range data applications, performs well in excess of 100 kilometers. These greater geographic ranges are also coupled with data rates of 200M bps. Telephone carriers have reported successful transmission of 0.5 gigabit over more than 100 kilometers of repeaterless single-mode fiber.

Connectivity—Optical fibers are most often connected in a point-to-point fashion. Multidrop configurations are possible; however, cost and reduction of topological flexibility make this less practical. Both passive and active star couplers are used to provide optical taps in bus topologies. They permit stations to be connected in a point-to-point star topology with the active star coupler functioning as a concentrator—similar to the approach used in a twisted-pair Ethernet environment. Unlike the active star coupler, the passive star coupler does not provide repeater functions.

The implementations most commonly encountered in office buildings today utilize optical fiber cable for trunks (risers that run between building floors) and sometimes for feeders on a single floor. Distribution to the less stable workstation environment is usually accomplished by using multiplexers and interfaces to twisted-pair or coaxial cable.

Noise Immunity—One of the most significant advantages of optical fiber is that it is essentially 100% immune to EMI and RFI. It requires no shielding and is completely impervious to any external interference. Attenuation is a less significant issue in optical fiber transmission, and losses are usually limited to about 0.05 to 0.07 dB per kilometer. One must be concerned, however, with the signal loss that occurs at taps and splices. This will vary with the nature of the tapping/splicing technology. Elastomeric splices yield the highest loss but are the least expensive. Optical connectors are more difficult to install and are more expensive but reduce the insertion loss. Fusion splices are the most expensive but yield the lowest insertion loss.

Security—Since optical fibers neither absorb nor emit radiation, they are largely secure from intrusions or tapping. Difficulty in tapping and in minimizing the insertion loss prevents most intruders from successfully penetrating or compromising the security of the system.

Installation Considerations—Optical fiber exhibits a number of positive attributes insofar as installation issues are concerned. The danger of electrical short circuits between conductors is a "nonissue," and the cable can even be totally immersed in water without affecting the signal. The cable itself is lighter than other media and facilitates installation. Fewer problems will be encountered with respect to building codes, but specific implementations must be verified for variable compliance issues.

Applications—Optical fiber has been used primarily for wide area computer and voice communications, but progress in the development of taps, splitters, and couplers has resulted in increasing use for intrabuilding and local area network implementations. Optical fiber alternatives are offered as part of most proprietary cabling systems, but one should compare the specifications of these optical fibers against industry standards. For example, early IBM cabling system implementations called for 100-micron fiber. This is not compatible, however, with the industry standard of 62.5 microns (EIA/TIA-568). The 100-micron fiber is superior for short-range implementations, such as IBM's token-ring, where repeaters are closely spaced, but degrades in longer runs. As an aside, the EIA/TIA-568 committee adopted the 62.5 um implemenation because of its popularity among the telephone companies and the fact that it had a very large installed base as opposed to the 100 um.

One should exercise caution in using fiber in unstable environments where user-connected devices are moved frequently. Expect to see extraordinary growth, however, in all intrabuilding applications for fiber. The dominant medium of the future will be optical fiber combined with unshielded twisted pair.

Cost—Optical fiber is somewhat cost intensive. The materials are more expensive than other media; however, this is often offset by greater capacity, reliability, and flexibility.

Standards and Proprietary Implementations

EIA/TIA-568 Intrabuilding Wiring Standard

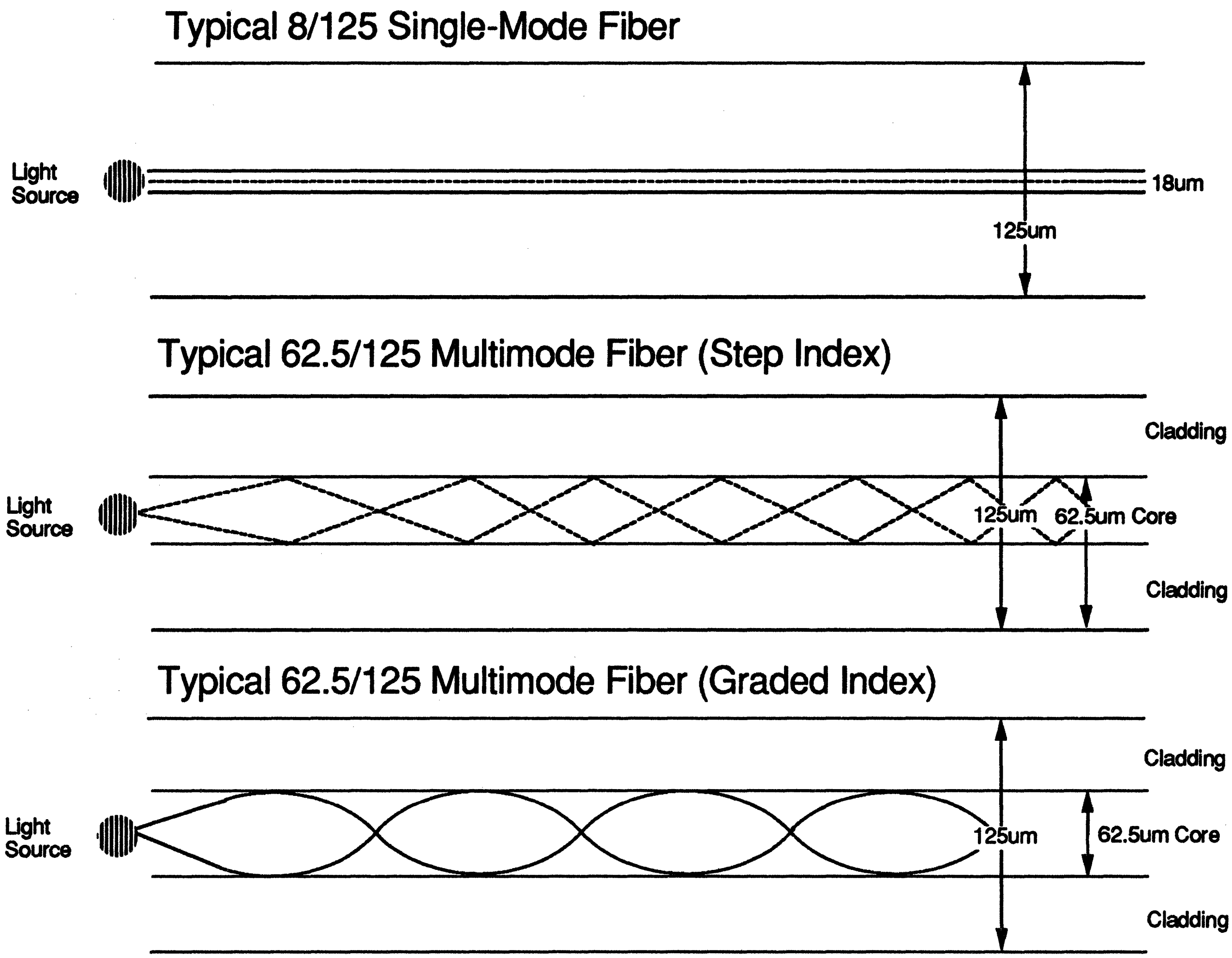
Introduction

EIA/TIA-568 is the product of over six years of intensive negotiation and compromise between major computer and telecommunications equipment vendors. The net result, while not perfect, is a major accomplishment. It provides planners with the tools to develop and install multivendor building wiring based on a flexible generic model—while using "off-the-shelf" products and technology.

Scope

The Telecommunications Wiring System Structure (TWSS) defined in 568 is intended to support up to 50,000

Figure 10.
Fiber Types



(a) Single-mode step index (b) Multimode step index (c) Multimode graded index.

individual users within 10,000,000 square feet of office space. The focus of this specification is on commercial office space. Industrial facility specifications are contained in another standards document.

Telecommunications Wiring System Structure (TWSS)

The standard includes detailed specifications on all aspects of the TWSS:

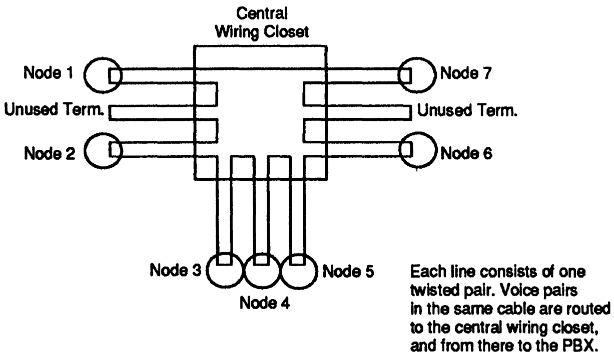
- Horizontal wiring—from the work area telecommunications outlet to the telecommunications closet.
- Backbone wiring—interconnection of telecommunications wiring closets.
- Work area wiring—from the telecommunications outlet to the station equipment.
- Telecommunications closet—area set aside to house TWSS equipment.
- Equipment room—similar to telecommunications closet but usually centralized and housing more complex equipment.
- Entrance facilities—telecommunications service entrance, including safety devices.

The 568 TWSS supports a variety of different cable types:

- Horizontal system
  - –4-pair 100-ohm unshielded twisted pair
  - –2-pair 150-ohm shielded twisted pair
  - –50-ohm thin RG-58 "thin" Ethernet coax
  - –62.5/125um optical fiber
- Backbone system
  - –multipair 100-ohm unshielded twisted pair
  - –150-ohm shielded twisted-pair cable
  - –50-ohm thick Ethernet coaxial cable
  - –62.5/125um optical fiber.

The 568 standard contains detailed specifications on the electrical and physical characteristics of each of these cable types. Connector specifications are included for all media; however, optical fiber connector specifications are incomplete as of this writing. They include mating durability data, but the jury is still out on the issue of whether the standard should be ST (straight tip) or biconic connectors—or both.

Figure 11.
Star-Based Ring



Star-based ring architecture using IBM cabling system.

IBM Cabling System

The IBM cabling system is an intrabuilding wiring scheme designed primarily to support the interconnection of IBM host computers, workstations, and PCs, with the added advantage of providing conventional telephone cable as an option for those sites intending to rewire for a PBX. It facilitates the implementation of a star-based token-ring, which lends itself to much easier maintenance and reconfiguration than traditional ring architectures. Figure 11 illustrates a typical star-based ring.

The cabling system includes all of the patch panels, connectors, faceplates, baluns, and any other hardware necessary for implementation. Table 1 presents the specifications for the various cable types currently included within the IBM cabling system.

IBM's "pre" cabling system environment included a variety of product-specific incompatible cable types, including twisted pairs, coaxial cable, and twinaxial cable. The cabling system provides common cable implementations for all of the above products.

Installing the IBM cabling system, or any cabling system for that matter, is most cost effective when organizations are required to rewire for a PBX. Installing the voice wiring can be leveraged to subsidize the data facilities—since the incremental cost of the IBM Type 1 or 2 data cables, plus the additional labor to pull these heavier cables, will almost always be significantly cheaper than if both data and voice cabling were to be installed separately.

AT&T SYSTIMAX Premises Distribution System

AT&T's SYSTIMAX Premises Distribution System (PDS) is a multifunctional wiring plan designed to support voice, data, graphics, and digital video communications on the user's premises. SYSTIMAX PDS comprises cables, cross-connects, adapters, electronics, and protective devices, all arranged in a hierarchical riser/distribution system. It encompasses optical fiber and unshielded twisted-pair copper media and is suitable for single-building as well as campus environments.

Current implementations of SYSTIMAX PDS employ 62.5/125-micron multimode optical fiber and 22/24/26 AWG unshielded twisted pair. AT&T has maintained the position that unshielded twisted pair will provide adequate performance for almost all voice and data requirements. The company contends that shielding confines the electric and magnetic fields, causing changes in induction, resistance, and capacitance, thus increasing attenuation. There are strong dissenting opinions, however, and other organizations will readily provide their own test results proving that shielding is a cost-effective mechanism. For example, IBM responds that shielding eliminates most disturbances (EMI/RFI) and that increased attenuation is a "nonissue" in the context of short token-ring wiring lobes.

SYSTIMAX PDS features six discrete subsystems:

1. Work location wiring system—a station mounting cord connecting a terminal or workstation to a universal wallplate.
2. Horizontal subsystem—connections from a panel in a satellite wiring closet to the universal wallplates in the work location.
3. Backbone or riser system—the riser cables interconnecting the different floors in a multistory building.

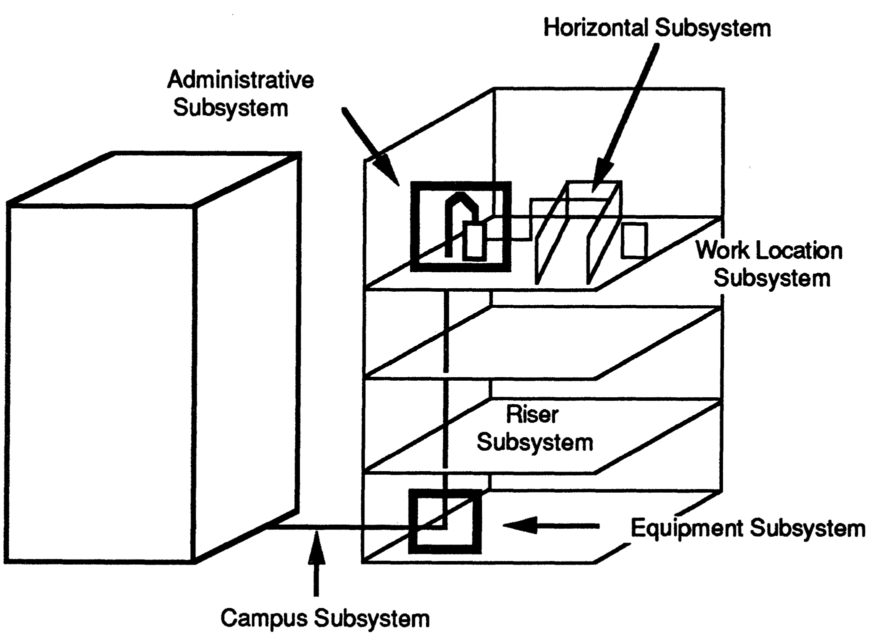


Figure 12.
AT&T SYSTIMAX Premises Distribution System

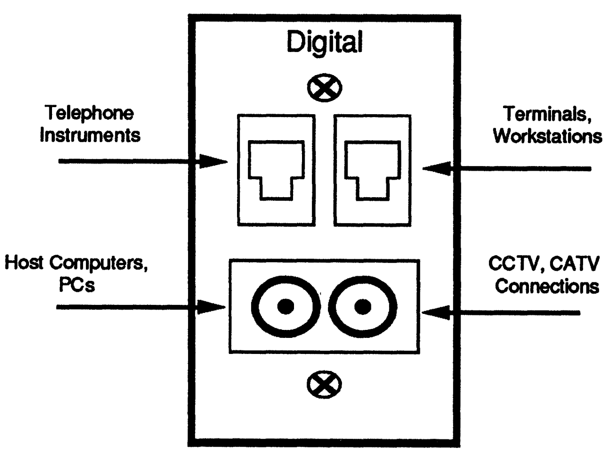
Table 1. IBM Cable Type Specifications

Cable	Specification Number	Description
Type 1	4716748	Two pairs of AWG #22 conductors with braided cable shield.
Type 1 Plenum	4716749	Two pairs of AWG #22 conductors with braided cable shield and Teflon or equivalent dielectric.
Type 1 Outdoor	4716734	Two pairs of AWG #22 conductors with corrugated metallic shield suitable for aerial or underground installation.
Type 2	4716739	Two pairs of AWG #22 conductors with braided shield accompanied by four pairs of AWG #22 telephone conductors and Teflon or equivalent dielectric.
Type 2 Plenum	4716738	Two pairs of AWG #22 conductors with braided shield accompanied by four pairs of AWG #22 telephone conductors.
Type 3	ANSI/ICEA S-80-576-1983 (Bell 48007)	Four twisted pairs of AWG #24 100-ohm telephone wire.
Type 5	4716744	Two 100/140 optical fibers suitable for indoor, aerial, or underground (in conduit) installation.
Type 6	4716743	Two twisted pairs of AWG #26 stranded conductors to be used as patch or jumper cables in wiring closet or from wall outlet to device.
Type 8	4716750	Two pairs of AWG #26 solid conductors in parallel for use under carpeting.
Type 9	6339583	Two twisted pairs of AWG #26 stranded or solid conductors.

4. Administrative subsystem—a frame or patch panel where circuits can be rearranged or rerouted. It includes cross-connect and interconnect hardware as well as modular jack assemblies.
5. Equipment cabling subsystem—that section of SYSTIMAX PDS that included the cables and connectors between equipment units and equipment rooms.
6. That section of SYSTIMAX PDS that includes the cabling between or among buildings within a campus or an office park.

Figure 12 shows the relationship between the six SYSTIMAX PDS subsystems.

Figure 13.
DECconnect Faceplate



DECconnect Communications System

The DECconnect communications system combines multiple technologies in one cabling system. Voice communications are provided via 100-ohm unshielded twisted pairs. Terminal communications requiring full- or half-duplex communications up to 19.2K bps are also provided on twisted pair. High-speed LAN applications employ 50-ohm thin Ethernet coaxial cable, sometimes affectionately known as "Cheapernet." Thick Ethernet cables are used to interconnect "Satellite Equipment Rooms" but are not generally used for work area implementations. This approach is compatible with the EIA/TIA horizontal and backbone approaches. In addition, DECconnect also provides video capabilities with the inclusion of 75-ohm CATV cable in the architecture.

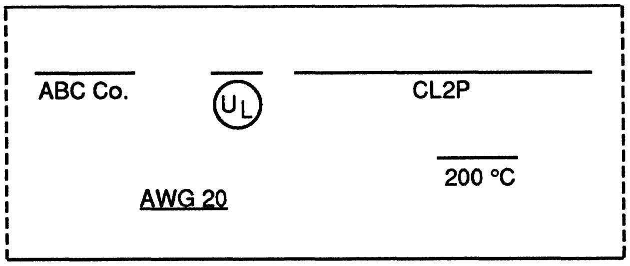
Common faceplates provide connections to each of the network facilities at each work area or office location. Figure 13 illustrates the configuration of the DECconnect faceplate. The faceplate provides a BNC coaxial cable connector for the Ethernet connection, a modified modular jack for terminal connections, a modular jack for telephone instruments, and an F-connector for CATV cable.

Digital has also been one of the foremost leaders developing Ethernet technologies. It provides a variety of active and passive components to interconnect thick, thin, and twisted-pair Ethernets as well as "dumb" terminal facilities.

Cable Plant Installation

Installing cabling systems and components is far from a simple task. In addition to a complete understanding of the technical and practical limitations of various media, one

Figure 14.
NEC Cable Markings



Where ABC is the cable manufacturer;
" AWG defines type of cable construction;
" UL is the UL trademark;
" CL2P is the circuit type identification code;
" 200 °C is the temperature rating.

must also be familiar with the national and local building and electrical codes. Methodologies for estimating the quantity and cost of the physical wire and components must be established. These may be manual, automated, or semiautomated procedures. The network architect must be able to perform a site survey to determine whether there will be any unusual installation difficulties. The physical cable plant must be designed with attention to the location of taps, splitters, amplifiers, and repeaters. Procedures for testing the medium and components before and after installation must be established.

While each installation will present its own set of specific issues and difficulties, this section of the report presents an overview of the above agenda.

National Electric Code (NEC)

The National Electric Code is a comprehensive set of regulations that significantly affects the communications industry. Note, however, that the NEC is essentially only a "model" code. While some local jurisdictions adopt it as published, others more stringently constrain installations. For example, while the NEC permits plenum cable in certain implementations, local codes may prohibit its use outside of conduit.

There have been a number of changes from the earlier 1984 edition involving new specifications for clearances, multiple "messengers" or cables on the same structure, optical fibers, and others.

The code is divided into several "articles:"

1. Article 725 addresses remote control signaling and power-limited cable.
2. Article 760 deals with power-limited fire alarm cable.
3. Article 770 provides specifications for optical fiber cable.
4. Article 800 covers various forms of communications cable.
5. Article 820 covers CATV cable specifications.

Copies of the NEC are available from the American National Standards Institute and the IEEE Standards Office.

Figure 15.
Cable Installation Estimating Form

Description	Materials			Labor		
	Qty.	Price	Extension $	Unit	Hours	Cost
2½" Conduit	550	2.0497	1,127.34	0.135	74.25	1,856.25
2½" Connections	14	10.60	148.40	0.45	6.3	157.50
2½" Couplings	55	8.79	483.45	—	—	—
2½" Elbows	4	11.1998	44.80	0.135	5.4	135.00
2½" Straps	91	0.75	68.25	0.75	68.25	1,706.25
RG59 Coax PVC	15,450	0.220	3,399.00	0.07	108.15	2,703.73
J. Boxes	6	56.75	340.50	0.01	6.0	150.00
Boxes	85	1.4569	123.94	0.50	42.5	1,062.50
Terminations	170	3.52	598.40	0.15	25.5	637.50
Plates	85	0.9210	78.29	0.10	8.5	217.50
Total Materials Cost			6,412.27	Total Labor		8,621.23

These prices and labor estimates are for illustration purposes only. They are provided only as a guide in the methodology of estimating, as opposed to representing benchmark cost.

Adapted with permission from "How to Specify Bid and Install Plenum Cable" © 1986–E.I. DuPont de Nemours & Co. (Inc.)

Procuring Cable

When procuring either twisted pairs or coaxial cable, it is important that the cable type, conductor size, insulation and jacketing materials, shielding, and cable classification be explicitly specified. In the case of many applications, coaxial cable in particular, the characteristic impedance (50, 75, or 93 ohms), the nature of the core dielectric, and the conductor material must also be specified. Be sure to consult computer equipment manufacturers' specifications for additional information. Exercise caution in specifying cable diameters since this will be important in selecting connectors, as well as estimating conduit capacity.

Manufacturers of Underwriter Laboratories (UL) classified cables are listed in the UL Green Book, which is available from the UL. All commercial cables for NEC 725, 760, and 820 must be UL listed. In addition, the UL conducts a "tunnel test" (the UL 910 Tunnel Test) for plenum cables. Some plenum cables bear markings that they are plenum listed or plenum approved; however, one should be sure that they are UL listed.

UL classified plenum cable identification bears markings every 24 inches in accordance with the NEC. Figure 14 illustrates these cable markings.

Installation Methodology

Site Surveys

The first activity that must be performed is the site survey. The network designer should physically walk through the site with the construction supervisor and user personnel and identify the locations of all equipment—host computers, workstations, PCs, telephones, and any other devices that may require access to the cable plant. Special attention should be paid to the locations and topology of floor troughs and inaccessible areas. All cable routes and tap locations should be clearly identified on the floor plans or blueprints.

Questions to be answered during the site survey include the following:

1. Where are the nodes located?
2. What is the impact of geographic dispersion?
3. Can the floor troughs or conduit handle additional cable?
4. How volatile is the environment? (Do users move frequently?)
5. Are there sources of electromagnetic interference?

Cable Plant Design

Design specifications and constraints vary for different types of LANs and telecommunications implementations. Before designing cable plant, it is important to verify equipment vendor specifications against standards such as the IEEE 802 LAN specifications.

Special considerations apply to "broadband" LANs. Design calculations must be performed to determine the "reference signal level"—the amplitude of the radio frequency signal that appears at each user outlet, which is measured at the network's highest frequency. In addition, the noise floor or minimum noise level of the system must be determined. This is a function of the ambient temperature, the characteristic impedance of the cable, and the channel bandwidth. Other calculations involve amplifier noise, system carrier-to-noise ratio, and splitter/combiner noise.

Broadband system design must ensure that signal strength is evenly maintained across the entire network. As an aid in calculation, dB should be used to express signal strength. The system will be easier to maintain if all amplifiers are identical and evenly spaced upon the cable. Losses, both insertion and attenuation, can then be equated to amplifier gain.

Other important guidelines in broadband system installation focus on ensuring that cable routes be evaluated to ensure that manufacturers' bend radius specifications are heeded. If a CATV cable is crimped in bending, the physical change in the relationship of the two conductors will impact the electrical properties of the cable.

Plan your CATV system as if you were going to maintain it daily. Even if reconfiguration is not that frequent, orderly installation will make your life much easier:

1. Clearly mark and band all drop cables at each end.
2. Attach visible markers to the ceiling to indicate tap, splitter, and amplifier locations.
3. Ensure that the proper size "F" connectors are used.
4. Route cables via the shortest possible distance.
5. Attach terminating resistors to any unused wall outlets.

Special test equipment may be required to design and maintain your CATV network. These include the following:

- RF spectrum analyzer—provides a graphic display of the frequency spectrum of the cable.
- RF sweep generator—serves as a signal source for testing and generates RF output signals from 5- to 500MHz.
- Field strength meter—a tuned RF voltmeter used for determining the amplitude of a signal at a specific frequency.
- Multimeter—used for checking power supplies and resistance.
- Cable reflectometer—used to precisely identify the location of breaks, kinks, or other faults in the cable.
- RF radiation monitor—sometimes called "sniffers" or "bloodhounds" and are used to detect leakage of RF signals at taps, splitters, and other connections.

Estimating Cable and Components

After you have performed your site survey and designed your cable plant, whether it be CATV, baseband coaxial cable, twisted pairs, or optical fibers, you are then ready to begin the process of estimating the materials and labor costs of the installation.

It is convenient to use a form as illustrated in Figure 15. The purpose of this form is twofold: first, to provide a checklist of materials, and second, as a budgetary estimate for the project.

Preinstallation Tests and Checks

A variety of tests and checks should be made on system components before installation. As a rule, users should undertake all tests that can be performed prior to installation:

1. Check cables while they are still on the spool. Using time domain reflectometry (TDR) techniques, one can determine whether there are breaks, crimps (in the case of coaxial cable), or other damage that induces electrical changes.

2. Check cable terminators using TDR techniques as well.
3. Transceiver and drop cables should be checked as well as the operation of transceivers and repeaters.

All of these tests are essentially easier to perform on a unit basis than on an integrated system. Imagine the alternative of trying to find a failed repeater or transceiver in a multistory office building after installation.

Postinstallation Tests and Checks

After installation, all cables and components should be visually inspected using TDR for backbone and trunk cables. The frequency spectrum of the system should be checked to isolate induced noise frequencies. In the case of coaxial cable systems, grounding should be checked using an ohmmeter.

Cable Plant Management

Cable management systems are usually freestanding, PC-based database systems that permit a user to track, manage, and maintain an inventory of cabling resources within a facility. These systems often provide graphics Computer-Aided Design (CAD) capabilities to view cable entities in the context of a floor plan. We use the term "computer-aided design" in a limited context, denoting the ability to develop electronic blueprints or floor plans employing high-resolution graphics capabilities and user interfaces.

The earliest of such systems were developed primarily to facilitate the management of telecommunications cabling resources associated with PBXs, but more recent entries address the specific issues of shared media LANs. Some systems provide CAD facilities to design a LAN within the context of a specific set of design constraints (such as IEEE 802.3 "Ethernet").

Hardware Components

Cable management systems reside on a variety of hardware platforms. For the most part, these are standalone PCs; however, they may be "networked" so as to enable distribution of the database resources in a client/server environment. A typical standalone hardware configuration includes the following:

- VGA graphics card with 640 x 350 resolution
- 80386 (80486) processor
- 2M to 4MB of memory
- Hard disk with approximately 5MB available for software (additional space will be required for database and CAD drawings)
- Mouse or pointing device
- Printer with wide carriage (or laser printer with landscape options)
- Optional plotter

Other components, such as time-domain reflectometers (optical or electrical), may be required to support the operating environment but are not integral parts of the system. TDRs operate on the principles of transmission line theory. If a cable is not terminated in its characteristic impedance, a reflection of the signal travels in the return direction from the cable end. A TDR generates signals on a cable, listens for reflections, and presents the operator with a readout on transmission parameters and distance of the fault from the point of measurement.

Software Components

Cable management software systems can manage a variety of wiring resources, including data, voice, LAN, and, in some cases, HVAC and control cables. Ideally, the software should perform the following functions:

- Cable and path identification
- Cable route presentation
- Assignment of cable facilities and routes
- Tracking conduit "fill" and usage
- Identification of unused cable facilities
- Equipment inventory for attached components
- Tracking service order activity
- Maintenance of trouble report database
- Management and operation reporting mechanisms

Cable management systems use a variety of database resources. These relational database systems provide superior flexibility in terms of the ability to construct different "views" of the information, using structured queries. In the case of a server-resident database in a multiuser environment, the system should also facilitate remote procedure calls (RPCs), thereby minimizing the amount of data to be transmitted and the processing requirements at the client workstation. Large organizations with several Telecommunications Service Center representatives will undoubtedly benefit from "multiuser" capabilities.

While some vendors develop their own proprietary database facilities, there are significant advantages to using off-the-shelf, de facto-standard systems (such as dBASE, RBase, etc.). These systems will enable users to create or download information from other sources to update the cable management system database. This facility will be particularly important when gathering information on existing cable/component resources for conversion to a new system.

Other desirable features include a report design/generator for custom user requirements. This might include interfaces to accounting systems for the purpose of "charging back" service costs to the appropriate cost center.

An essential element of any cabling system is label generation. Cables, faceplates, connection points, and connectors must be labeled with specific circuit identification codes. This is imperative for both installation and troubleshooting any system. Labels should be generated by the system after the design is finalized, but *before* the actual installation process begins.

Most cable management software systems can interact with digitized mapping systems using CAD technolcgy.

Most cable management software systems run in either a DOS, MS-DOS, or AIX/UNIX environment.

Selection Guidelines

There is a variety of highly specific functions and capabilities available within the realm of existing products. While the requirements will vary depending upon the environment and "culture" of the user organization, the following breakdowns are offered as a checklist to assist in evaluating existing cable management products.

- Proprietary cabling architecture support (e.g., AT&T SYSTIMAX/PDS, IBM Cabling System, DEConnect, etc.).
- Ability to present complete cross-connect paths of all cable pairs.
- Ability to maintain backbone vertical riser and horizontal distribution records.
- Maximum number of cable records supported: a 10-floor building with 2,000 people, each with a 4-pair termination, will require a minimum of 8,000 cable records. This does not include preexisting 25-pair runs or growth.
- Number of buildings to be supported.
- Component/Equipment records (e.g., vendor, model, serial #, description, date installed, user/owner, etc.).
- The ability to charge for cabling resources by department or cost center.
- Preparation of labels for cable installation.
- Ability to validate "proposed" cable runs against wiring specifications (distance, signaling rate, attenuation, etc.).
- Report Generator for user-customized reports.
- Vendor installation and training.
- Vendor operational support (on-line hardware and software).

Ongoing and Emerging Developments

The world of intrabuilding cabling is constantly changing. Major efforts undertaken in recent years have since become realities: the EIA/TIA-568 standard, the FDDI optical fiber token-ring, and expanded IEEE Local Area Network Specifications. New frontiers have opened in the following areas:

- The American National Standards Institute is continuing development of the FDDI specification. It is expected that FDDI-II will provide for integration of data and isochronous voice communications.
- The IEEE has developed bridge specifications which greatly enhance the flexibility and geographic range of LANs. Initial specifications address the implementation known as "Spanning Tree Bridges," which lend themselves to bus-oriented LAN environments. Future work may include the development of source routing bridges, which are generally preferred by the token-ring proponents. It will be necessary to ensure that both bridge architectures provide for interoperability.
- Wireless LANs are presently generating interest. Most early implementations are Radio Frequency based and, as such, signals can penetrate walls within a reasonable range. Products are available on the market today and provide repeater capabilities so they can be connected to standard backbone LANs.
- The EIA/TIA will continue development of the 568 specification. Issues remaining to be resolved include the following:
  - –Evaluation of connectors for optical fiber.
  - –Inclusion of additional media choices.
- The IEEE will keep working on the use of optical fiber for LAN environments, interfaces to ISDN, and the initial work in the area of wireless LANs. ■

This report was prepared exclusively for Datapro by Michael L. Rothberg. Mr. Rothberg is president of Applied Network Solutions, Inc., a Somerset, NJ firm specializing in designing, developing, and implementing local and wide area computer networks for government and commercial clients and providing market and product planning services for communications suppliers.

Prior to founding the company in 1981, Mr. Rothberg was a vice president of the Chase Manhattan Bank, N.A., where he pioneered the application of digitized speech and local networking technology to support banking applications. He is a frequent contributor to trade publications and is a member of the Program Advisory Board of the Interface Exhibitions and Conferences.

Future Wiring Considerations: Fiber vs. Twisted Pair

[redacted]

In this report:

Datapro Summary

The cost of installing network wire is high, which is precisely the reason that two wiring technologies—twisted pair and fiber—are being closely examined by LAN managers. The most popular cabling option of five years ago, coaxial, is nearly obsolete now. The breakneck pace of the networking industry makes it imperative for managers to plan for future wiring configurations now.

Consider the eternal wiring question: Which physical medium is most economical to install today to serve both current and future data needs?

That question is always of interest because the price of installing wire can be a major component of a network's total cost.

This report examines whether users should install unshielded twisted pair today to support future speeds ranging from 16M to 100M bit/sec.

What may affect that decision are prospects that future networks may operate at speeds far above 100M bit/sec. At the June International Conference on Communications in Denver, an IBM representative is scheduled to present a paper on the performance of Fiber Distributed Data Networks at 1 G bit/sec.

If users have inklings that they may require such speeds in the foreseeable future, they should probably install fiber. While there are questions about whether unshielded copper will be able to support 100M bit/sec transmission for any reasonable distance, the consensus of experts holds that at rates above 100M bit/sec, only fiber will do. (For an evaluation of cabling choices, see Figure 1.)

Not discussed here is the alternative of using the new radio local-area networks to replace copper. The issue is solely unshielded twisted pair vs. shielded twisted pair vs. fiber. Coaxial cable, a medium used by practically all LANs five years ago, is rarely being installed these days.

OK for Ethernet

It is generally accepted that unshielded twisted pair is an adequate medium for Ethernet's 10M bit/sec data rate. This is noteworthy because two years ago, questions arose regarding whether the new Ethernet-over-unshielded twisted-pair specification, 10BaseT, could meet Ethernet's specified low error rates and the Federal Communciations Commission's anti-radio-emission requirements.

It's also worth noting the limitations of 10BaseT as compared with traditional Ethernets. Ethernets can use a single thick coaxial cable for distances as long as 1,650 feet, or they can employ more flexible thin-wire coaxial for up to 610 feet. The IEEE's 10BaseT specification mandates two unshielded twisted-pair cables and a distance of no more than 100 meters (about 328 feet) between any two transmitting points—workstations, repeaters or active hubs in a wiring closet.

For campus networks, therefore, 10BaseT is not used ·over unshielded twisted pair. For longer distances, 10BaseT

This Datapro report is a reprint of "Weighing Factors in the Twisted-Pair/Fiber Choice" by Edwin Mier, pp. 1, 29, 43, and 57-58 from *Network World*, Volume 8, Number 17, April 29, 1991. Copyright © 1991 by International Data Group. Reprinted with permission.

Figure 1.
Competing Designs for 100M bit/sec Data Over Twisted Pair

Company	Modulation	Cabling/Distance	Signal Power/ Loss Characteristics	Advantages	Disadvantages
IBM Armonk, N.Y. (914) 934-4000	125M-baud signal using standard FDDI 4B/5B encoding; primary radiated emission at 62MHz	Requires IBM Type 1 shielded twisted pair; up to 100 m	Transmit signal level at 200 milliVolts (mV); receive signal level very low, from 40 mV (14 dB loss) down to 20 mV (20 dB loss)	–Should enable existing FDDI chipsets to be used –Low power signal; should satisfy FCC and international emission limits	–Very sensitive receiver circuitry required; could raise product costs –Requires IBM Type 1 shielded twisted pair
SynOptics Communications, Inc. Mountain View, Calif. (408) 988-2400	Same as IBM above; radiated emissions also peak at 62MHz	Requires IBM Type 1 shielded twisted pair; up to 100 m	Transmit signal level is 800 mV; receive signal level from 160 mV (14 dB loss) down to 80 mV (20 dB loss)	–Should enable existing FDDI chipsets to be used –Likely the least expensive design to implement	–Concern about meeting FCC emission limits under new test guidelines or international standards –Requires Type 1 shielded twisted pair
Crescendo, Inc. Sunnyvale, Calif. (408) 732-5955	Proprietary encoding method not FDDI-compatible; radiated emissions peak at 31MHz	May operate over unshielded twisted pair at limited distances (around 50 m)	Transmit signal level 850 mV; receive signal level 300 mV (9 dB loss over 50 m)	–Enables use of unshielded twisted pair wiring over short distances –Lower frequency emissions more likely to pass FCC	–Proprietary encoding not supported by existing FDDI chipsets; could raise costs dramatically –Emission characteristics on unshielded twisted pair need further study

hubs employ an interface to coaxial or fiber. At most user sites, however, 100 meters over unshielded twisted-pair segments is considered adequate to let users reach their nearest telecommuncations wiring closet.

The 10BaseT specification doesn't recommend any particular type of twisted pair. Even old, already-installed analog phone wiring—usually 24 American Wire Gauge (AWG), but sometimes the smaller 26 AWG—is considered usable. However, to be on the safe side, it is best not to attempt 10BaseT over pairs that may have lost their twists.

Regardless of the gauge of twisted-pair wire used, the node-to-hub segment should support the specified bit error rate of 10^9; 10BaseT calls for an end-to-end system bit error rate of 10^8.

The possibility exists that some 10BaseT installations may be emitting too much electromagnetic energy in the form of radio waves. A fundamental problem of LANs is that at high data rates, with even moderately low electrical power, unshielded wiring turns into a broadcasting antenna. To avoid that, new higher speed LANs that rely on unshielded twisted pair reduce signal power even more, requiring increased receiver sensitivity but also usually decreasing distances.

Despite that, some experts believe that—depending on the type of unshielded twisted pair and the connectors and hubs used—some 10BaseT installations may be emitting excessive electromagnetic radiation.

The FCC sets rules to prevent that from happening. But according to Bill Aranguren, a development supervisor with AT&T Computer Systems, the FCC's current emission test policies and procedures are very vague. It is left up to manufacturers of 10BaseT equipment to determine on their own whether their systems' electromagnetic emissions are within acceptable limits.

However, a new FCC emissions test procedure, which will take effect in June 1992, includes much more specific—and stringent—measurement requirements. According to Aranguren, several 10BaseT hub products now on the market would not pass these new test guidelines. (Because these allegations could not be independently confirmed, it was decided not to list the name of those products.)

Aranguren also raises the issue of whether any large 10BaseT network employing unshielded twisted pair is possible using a single active hub "I question whether any (manufacturer of hubs or concentrators) with 100 or more ports and that measures in accordance with the new procedure could pass (the new FCC requirements)," he says.

Unshielded Token Rings

If electromagnetic emissions from twisted-pair LANs are excessive at the 10MHz signaling rate of 10BaseT, they are likely to be worse at the higher signaling frequency of a 16M bit/sec token ring. It is mainly for this reason that IBM does not endorse operation of 16M bit/sec token rings over unshielded twisted-pair wire.

In the mid-1980s, four years before unveiling the 16M bit/sec token ring, IBM specified what is now known as IBM Type 1 cable for data rates of 16M bit/sec and higher—actually, for rates up to the 100M bit/sec range. In diameter and weight, Type 1 more closely resembles the coaxial cable used in 3270 terminal networks than the more familiar telephone twisted-pair-wiring.

Type 1 consists of two pairs of twisted, solid 22-AWG copper conductors, surrounded by a metal shield all encased within a polyvinyl chloride jacket.

The Type 1 cabling is also characterized by a higher impedance (measured in ohms of resistance) than telephone-type twisted-pair wire.

As a general rule, the greater the impedance, the less a highspeed data signal attenuates. IBM-specified cabling, which is made by several manufacturers, is 150 ohm—meaning it offers 150 ohms of resistance. Most telephone-type pair is 100 ohm.

New Twists to Twisted Pair

Network users' need for highspeed data transmission has caused wire manufacturers to add a new twist—several, in fact—to twisted-pair wiring.

A new type of data grade, or high-performance, twisted pair is available from leading cable manufacturers such as AT&T and Belden Wire and Cable Co. Designed for data speeds of 16M to 100M bit sec, this wiring costs several times more than the standard nickel-a-foot telephone wiring but less than IBM Type 1 shielded twisted-pair cabling.

What makes this new twisted pair truly different from the old is the way it is twisted.

Like most telephone wiring, high-performance twisted pair is usually 100-ohm, 24 American Wire Gauge (AWG) with four pairs per cable. But unlike standard D-type inside wiring, 24-AWG cable, in which each wire pair is twisted about once per foot, high-performance wiring is twisted as many as eight or nine time per foot.

Quieting Cross Talk

Twisting wire minimizes radiated signal energy and ameliorates the effects of cross talk. Wire manufacturers have found that increasing the twists of twisted-pair wire more dramatically reduces the emissions and the wire's susceptibility to cross talk.

The number of twists per foot is specially calculated to support high data rate frequencies, up to the 125M-baud signaling rate of Fiber Distributed Data Interface.

According to Bill Aranguren, a development supervisor with AT&T's data-grade unshielded twisted-pair wiring, called Type 1061A, adds yet another feature: Each pair within the cable is twisted at a different twist rate.

AT&T also offers a teflon-coated plenum version, called Type 2061A. Shielded versions of both these cables, called Types 1261A and 2261A, for plenum, are also available.

Belden Wire and Cable calls its equivalent wire type 1455A and 1457A for unshielded twisted pair and Types 1456A and 1485A for shielded twisted pair.

Wire manufacturers are hesitant to publish hard data about the performance improvement of the new twisted-pair wire over the old, generally because the characteristics of the new twisted pair varying significantly depending on the level and frequency of the transmitted data signal.

But whether this new and improved unshielded twisted pair will be able to carry FDDI data rate signals and still meet Federal Communications Commission emission restrictions is the subject of some controversy today. In all likelihood, users will never find out.

The shielding inherent to Type 1 cabling reduces emissions to a fraction of what they would be without shielding, says Robert Love, senior engineer of IBM's Cabling system Center in Research Traingle Parck, NC. Love says that even the best grade of unshielded twisted-pair wire is still at least twice as lossy as Type 1 shielded twisted pair.

But he concedes that there is considerable user pressure for a new IEEE specification that allows 16M bit/sec token-ring operation over shielded twisted pair.

And so IBM is reviewing the situation. "We're looking at a basic design (for 16M bit/sec token ring on unshielded twisted pair) in the 1992 time frame," Love says. But he refuses to elaborate on the matter.

With this in mind, should users that are installing wiring now and project 16M bit/sec requirements in the future stick with Type 1, or should they shift to unshielded twisted pair? There are good reasons for sticking with Type 1 unless the user is willing to invest in one of the new high-quality varieties of unshielded twisted pair (see the sidebar "New twists to twisted pair").

IBM's new version of the 16M bit/sec token ring, which may be called 16BaseT, will almost certainly be more restricted—as far as the number of nodes and distance—than the current shielded twisted-pair-based 16M bit/sec token-ring specification.

Furthermore, it is unlikely that the new version can be supported on already-installed telephone wiring. One of the newer types of high-performance twisted pair will probably be required.

The Chances for TP-DDI

It is unclear whether data can be reliably and legally transmitted over twisted pair at 100M bit/sec, the data rate specified in the FDDI standard. Numerous competing technology proposals, each with unique benefits and problems, are currently being considered (see Figure 1).

The proposal from twisted pioneer SynOptics Communications, Inc. of Mountain View, Calif., calls for a transmitted signal similar in nature to FDDI's; that is, a 125M-baud signal pattern, which carries 100M bit/sec of data in the same 4B/5B encoding scheme that FDDI uses over optical fiber.

According to Peter Tarrant, SynOptics' FDDI product manager, the company is still working on developing a product that incorporates this proposal.

But some major participants in the 100M bit/sec-over-twisted-pair fray—referred to as Twisted Pair Distributed Data Interface (TP-DDI or TDDI) in the ANSI X3T9.5 standards committee that is addressing the issue—aren't convinced that SynOptics' design will work.

Unquestionably, 100M bit/sec data can be carried over twisted pair for short distances. The sticking point is whether this approach will generate an amount of radiated electromagnetic energy that exceeds FCC emission limits. If so, this approach would constitute an unlawful source of electromagnetic interference and perhaps even an environmental or health hazard.

"We have tested and established compliance with current FCC requirements," says SynOptics' Tarrant. He would not speculate, however, as to whether the current SynOptics' design will pass under the new and more stringent test procedures that go into effect next year.

Figure 2.
Comparison of Cabling Costs

Media	Termination Time per Connection	Cable Cost (per foot)	Cost per Wall Plate	Typical Installation Cost for a 100-Node LAN (complete)*	Installation Cost (per station)
Unshielded twisted pair; DIW-24 cable, polyvinyl chloride jacket, RJ-45 connectors	5 min. (4 conductors)	5 cents	80 cents	$88,000	$880
Shielded twisted pair; IBM Type 1 cable; IBM data connectors/wall plates	15 min. (4 conductors and ground)	22 cents	$9	$120,000	$1,200
Optical fiber; 62.5/125-micron multimode 'duplex' (single pair) fiber; multifunction wall plates	30 min. (2 fibers with ST connectors)	60 cents	$20 to $30	$180,000	$1,800

*Complete installation cost for two-pair copper or two-fiber (single fiber pair) to each station. Includes labor for cable pulling/termination, wall plates and patch panels (of fiber splice cabinets). Assumes PVC-insulated cabling—no plenum or Teflon. Costs will vary with location, due mainly to labor costs and code requirements. These comparisons try to show just the salient differences based on media selected and presume all other factors are nearly equal.

A Low-Power Alternative

IBM's TP-DDI design is similar to SynOptics' but it calls for a much lower signal power level, both for transmission and reception. Where SynOptics' design calls for an 800-millvolt transmit level, IBM has cut that down to 200 millivolts, a very significant difference in signal power.

The purpose of the IBM power cutbacks is to minimize emitted radiation, but it also requires that receivers be far more sensitive, which could also cause them to be much more expensive.

It's also unclear whether reliable receivers for such low-signal power levels can be commercially produced and, if so, how much they would cost.

In both IBM's and SynOptics' approaches, the transmitted signal radiates at a frequency of 62.5 MHz. This represents a much higher frequency emission than those of earlier twisted-pair LANs, the primary emissions of which have always stayed below 30 MHz.

The FCC considers emissions over 30 MHz as genuine interference threats to other forms of radio communications.

For example, the 62.5 MHz primary emission of both IBM and SynOptics falls within the range of frequencies used for regular VHF television broadcasting, and the potential certainly exists for interference between the two.

To minimize their emissions, both IBM and SynOptics require that only Type 1 twisted-pair cabling be used in their respective TP-DDI schemes. Furthermore, they require that two strands of Type 1 be used for each direction of the FDDI.

The full FDDI specification calls for dual counterrotating rings (on two pairs of fiber), so a full TP-DDI implementation would require four strands of the bulky Type 1 cabling. There are also proposals on the ANSI X3T9.5 TP-DDI schedule for a way to lower this emitted frequency by changing the type of signal encoding.

One such proposal from Crescendo, Inc. of Sunnyvale, Calif., calls for a proprietary 4-to-1 encoding technique to keep the primary radiated emission down to about 31 MHz. The company predicts that the scheme could yield 100M bit/sec data rates that would operate over unshielded twisted pair at limited distances—perhaps 50 meters.

The major criticism of Crescendo's approach, or any encoding technique that doesn't conform to the FDDI standard, is that it is incompatible with the FDDI chipsets that several suppliers, including AMD Co., Motorola, Inc. and National Semiconductor Corp., currently offer.

New more expensive chipsets would be required to produce 100M bit/sec signals in the new format.

Fiber Inevitable?

The problems with running data at 100M bit/sec and higher over twisted pair make it clear that twisted pair's days as the data network cabling of choice are coming to an end.

If twisted pair can handle 100M bit/sec applications—and that's still a big if—it will be limited to very short distances. And it will likely require a special grade of twisted pair, such as the IBM-specified cabling mentioned above. This cable is not much cheaper to install and maintain than optical fiver (see Figure 2).

As the table shows, the cost of a new 100-node LAN installation today may be doubled just by using optical fiber rather than regular twisted pair—from about $900 per station for two- to four-pair D-type Inside Wiring-24 twisted pair to $1,800 per station for a single pair of 62.5/125 micron multimode fiber. With Type 1 shielded twisted pair, the savings of copper over fiber is nearly halved.

Also, the cost of fiber and its installation are rapidly dropping. As they do, the incentive for perpetuating usage of copper twisted pair is quickly eroding.

Figure 3.
Cabling Selection: Making the Right Choice

Media:	**In-Place Unshielded Twisted Pair** 24 to 26 AWG	**New Unshielded Twisted Pair** D-type inside Wiring-24 (24 AWG)	**Data-Grade Unshielded Twisted Pair** AT&T 1601, Belden Wire and Cable Co. 1455A	**Shielded Twisted Pair** IBM Type 1	**Optical Fiber** 62.5/125-micron multimode
Pairs/Station:	Minimum of 2; often 25-pair telephone company cable; already installed	Minimum of 3; 4 recommended	4-pair	2-pair	1-pair
Application:					
Analog PBX/voice; modem/dial data; telecommunications	○	○	●		
Digital PBX/voice; ISDN; data/voice; 64K bit/sec. to T1	◍	○	○	●	●
LAN; up to 4M bit/sec.; low-speed token ring	◍	◍	○	○	●
LAN; up to 10M bit/sec.; 10BaseT Ethernet	◍	◍	○	○	●
LAN; up to 16M bit/sec.; high-speed token ring		◍	◍	○	○
LAN; up to 100M bit/sec.; FDDI data rates			◍	◍	○
Beyond 100M bit/sec.					○

○ A good choice; recommended ◍ May be technically possible, but not recommended ● Unnecessary; likely overkill

AWG–American Wire Gauge

In fact, even SynOptics, which owes much of its success to the continued exploitation of twisted pair for LAN data cabling, acknowledges that its as yet unnamed TP-DDI product will have a limited number of users. The product will be oriented only to users that already have Type 1 cabling installed and that plan to migrate from 16M bit/sec token ring to 100M bit/sec in the near future, a company spokesman says.

The Narrowing Options

Clearly, unshielded twisted pair has become the cabling of choice for Ethernet. New Ethernet installations are usually 70% unshielded twisted pair, and that percentage is rising, says John Mchale, president of Dallas-based NetWorth, Inc., which manufactureres a line of 10BaseT Ethernet hubs.

McHale says that 20% of new installations are running Type 1. The remaining 10%, he says, are running either coaxial cable (primarily as a distance extender to existing thin-wire Ethernets) or optical fiber.

Simply Impossible

In McHale's view, unshielded twisted pair is a simple impossibility for data rates of 100M bit/sec. It "clearly violates FCC rules," he says. "You would have to change the 4B/5B encoding [of FDDI], which would require new silicon."

Perhaps the best advice for choosing the right type of data cabling is summarized in Figure 3, provided by Trenton, NJ.-based East Coast Corp., one of the few wiring contractors that handles and installs both optical fiber and the gamut of unshielded twisted-pair cabling.

"There is increasingly a mix [of optical fiber and copper] going in," says Frank Kowalik, field installation supervisor with East Coast. "Fiber now is running, at least to the wiring closet, in most new state-of-the-art installations. And [fiber] is just now starting to go, along with copper, to the desktop."

According to Kowalik, a large portion of installation bids for new cabling and proposals that companies issue continue to specify only copper cabling.

But times are changing.

"It's a different cabling world today," he says. "Everyone wants their data network to handle future requirements, and many are realizing that unshielded twisted pair won't take them as far into the future as they would like." ■

Managing Data on PC LANs

In this report:

Datapro Summary

Initially, microcomputers came in through the back door at many corporations. PCs first captured the attention of techies intrigued by the challenge of squeezing meaningful information into 32K bytes of random access memory (RAM). At the time, MIS managers dismissed the devices as nothing more than toys, such as electronic calculators, that were incapable of having an impact on day-to-day operations. But microcomputers quickly gained power. Users now have 16M bytes of RAM on their desktops and are demanding even more memory. In many firms, these former corporate toys have been transformed into the foundation for strategic applications.

The Early Years of Microcomputers

Because microcomputers came in through the back door, most corporations never outlined formal procedures or application standards. As corporations search for ways to improve their use of computers, they find that they have purchased millions of dollars of equipment that cannot exchange even rudimentary information. They want to tie their computers into complete systems capable of sharing processing as well as data. Client/server computing is the current buzzword that describes attempts to this problem.

Much of the responsibility for transforming client/server from a buzzword into a tangible asset falls under the aegis of MIS managers, some of whom are intimidated by microcomputers and feel more comfortable dealing with large systems, such as mainframes. Once these managers take a close look at the company's microcomputers, they often discover chaos resulting from years of uncontrolled user purchases.

—By Paul Korzeniowski

These managers must somehow bring order to a perplexing array of different machines, applications, and local area networks. Making the problem worse, they find that tools to pull all these computers into a coherent entity are not readily available. Despite these challenges, there are signs of hope and steps that managers can take to simplify the process of pulling different systems together.

To understand the task that managers now face, one must retrace the steps taken in most large corporations. When microcomputers first appeared, they did not have sufficient processing power to support important applications, nor did the A: prompt offer any significant productivity improvement compared to existing devices such as IBM 3178 terminals.

Microcomputers also lacked the security and backup features with which MIS managers had become familiar. Consequently, PCs were easily dismissed as executive tinker toys incapable of supporting important work. When IBM unveiled its personal computer 10 years ago, the company estimated that 250,000 of the devices would be sold during its five-year life cycle. Engineers and scientists were thought to be the most likely users. The market was deemed so trivial that IBM asked Microsoft to write the basic operating system.

Middle Managers Implement PCs

However, microcomputers soon became an integral part of many middle managers' sets of tools. The first product to generate significant interest in microcomputer applications was Lotus Development's 1-2-3 spreadsheet. The product was based on work done by Visicalc, a small company that designed a spreadsheet program for Apple II computers. Unfortunately for Visicalc, the company concentrated its research and development efforts on Apple computers. Lotus concentrated on the nascent IBM microcomputer market and found that businesspeople felt more comfortable dealing with IBM than Apple.

Because 1-2-3 automated some tedious procedures (e.g., creating department budgets), its use spread rapidly. When financial analysts began to use 1-2-3 to do rudimentary number crunching, the executive market for PCs took off. PCs were purchased at such a rapid rate that in just its first sales year the IBM Personal Computer surpassed IBM's original five-year, 250,000-unit projection.

At the time, users were not interested in mainframe security and gave little thought to system backup (at least until their first disk crash). Many managers were responsible for budgets in the hundreds of thousands of dollars, so justifying the $2,500 purchase price of a microcomputer was fairly simple.

Microcomputer Usage Spreads

Once spreadsheets became popular, other applications for microcomputers developed. WordStar carved out a thriving business by offering a microcomputer word processing package. The product operated on a user's desktop, so he or she did not have to wait for a central system, such as a minicomputer or a mainframe, to complete the steps needed to produce a document.

Next, users began to tinker with microcomputer database management systems (DBMSs) and built applications to solve rudimentary department chores. For instance, sales executives began to store customer files in a database management system. Ashton-Tate became the most successful database supplier and the firm's dBase product quickly captured more than a 50% market share.

In general, however, PCs were purchased without any guidance or supervision from central MIS managers, who continued to dismiss microcomputers as something less than serious computing devices.

Limitations of Standalone PCs

As users worked with spreadsheet, word processing, and database programs, new requirements emerged. A financial analyst would often collect information that his or her supervisor needed to review. Initially, the review process was simple—the employee would walk to the supervisor's office and place a diskette in the B: drive of the supervisor's computer. This procedure was dubbed "sneakernet" because users ran from office to office to exchange information.

Output is another area that illustrated standalone microcomputer limitations. If five executives bought five computers, they also had to purchase five printers to print documents. Since each individual printer was used so little, it made sense to share printers. By sharing, executives could split the cost of expensive peripherals.

Managers began to search for better ways to share information and peripherals. At this point, LANs were emerging as a means for midrange computers to share information. As the market changed, however, LANs became the primary method to link microcomputers. (This is one reason the growth rate for midrange computer sales has slowed during the past few years.)

LAN Technologies

In the early years of the local area network market, a number of LAN technologies were available, and there was tremendous debate about which would emerge as the dominant means for connecting computers. Digital Equipment, Xerox, 3Com, and others advocated Ethernet technology, which uses broadcast techniques to move information over LAN wiring. IBM eschewed Ethernet in favor of token-ring technology, which requires a workstation to possess a network token in order to transmit information. Wang Laboratories and Sytek pushed broadband networks, which rely on bulky CATV cable but can support several signals over a single cable.

Wiring is only one part of the LAN equation, however. Users need a LAN operating system capable of moving information from one microcomputer to a second. In the middle 1980s, a handful of companies emerged to fill this void. Novell eventually became the market leader with its NetWare LAN operating system. Initially, NetWare was designed to simply connect workstations to printers and perform other rudimentary functions needed in small workgroups. It was not designed to connect all users in a corporation.

The Microcomputer Explosion

In the mid-1980s, the microcomputer market began to gain momentum. Microcomputer power increased at a rapid rate and users became accustomed to new machines being introduced every 18 months that doubled processing speeds and the amount of internal memory. Apple introduced the Macintosh, featuring a graphical user interface (GUI) that was a significant improvement over the interface provided by IBM.

The Macintosh also gave rise to a new application genre, graphic-based applications. Desktop publishing systems became popular. These products transformed microcomputers and printers into sophisticated systems capable of producing newsletters, manuals, and even books.

The additional processing power became the foundation for a new generation of software programs. Borland International quickly emerged as a market leader by selling programming tools and languages. Later, Borland entered the DBMS market with Paradox, a simple but fast DBMS. DataEase International also became a force in the DBMS market. The company's product, also called DataEase, offered a simple foundation for building applications and quickly became a market leader. WordPerfect became one of the leading word processing suppliers. The company originally designed software for midrange computers, but moved into the microcomputer market in the early 1980s. Microsoft began to challenge Lotus 1-2-3 spreadsheet applications with Excel, software originally designed for the Macintosh. Excel gained momentum as sales of Microsoft's Windows operating system began to take off.

The result—microcomputer data is generated by a hodgepodge of packages from a variety of suppliers. Sharing information is difficult because the suppliers support different file systems.

The LAN Explosion

As new products were making their way into the market, sales of local area networks began growing at an astronomical rate. Ethernet became an industry standard, first making its way into the midrange market with the help of companies such as Digital Equipment Corp., then gaining a foothold in the microcomputer market via vendors like 3Com.

Almost immediately after announcing its token-ring networking line in 1985, IBM decided to double production shipments. The company doubled the rate a second time and still demand outstripped supply. For more than two years, IBM was in the enviable position of trying to play catch-up with a runaway market.

As corporations gobbled up LANs, another set of issues began to emerge. After installing a LAN, departments quickly realized they needed information from other departments in the company. Thus arose the desire to interconnect multiple LANs within the same company.

Operating System Limitations

As companies began to examine these issues, shortcomings with microcomputer products became evident. Microsoft's MS-DOS operating system is not designed to run on a network and has many limitations. MS-DOS is single-tasking, meaning only one function can be completed at a time. It addresses only 640KB of memory, which is insufficient to support new applications. A user can layer Microsoft's graphical user interface (Windows) on top of MS-DOS, but the operating system's memory limitations makes Windows a weak competitor to the Apple Macintosh interface.

LAN operation systems, such as NetWare, also pose problems. The NetWare operating system does not take full advantage of the 32-bit processing power of microprocessors like the Intel 80386. Also, the operating system is designed for self-contained LANs and lacks features, such as distributed directory services, needed to connect a variety of LANs.

Unlike Novell, Banyan Systems designed a LAN operating system for an entire corporation. Banyan's VINES operating system includes advanced features, such as distributed directories, that make it a strong foundation for enterprise-wide networking. Unfortunately, Banyan has been unable to transform VINES' technical attributes into sales dollars; its market share (around 7%) is dwarfed by that of Novell (approximately 60-70%).

OS/2

With these limitations as a backdrop, IBM and Microsoft unveiled the OS/2 microcomputer operating system in May 1987. OS/2 was designed to overcome MS-DOS limitations. OS/2 has multitasking features, can address up to 16MB of RAM, and takes advantage of 32-bit processing. In addition, OS/2 was designed to run on LANs. IBM and Microsoft took base LAN technology, added enhancements, and announced two products: IBM's LAN Server and Microsoft's LAN Manager. Because the products had a number of features and were endorsed by the two heavyweights, many observers questioned whether Novell would be able to maintain its dominant market position.

Novell responded a short time later with a new release of its LAN operating system—NetWare 386. NetWare 386 included more processing power and better connectivity to microcomputers, such as the Apple Macintosh. NetWare 386 still did not have all of the functions desired by users, but it was able to maintain the company's lead in the LAN operating system market.

Eventually, Novell's position was helped by a falling out between IBM and Microsoft. When OS/2 was unveiled, IBM viewed it as the microcomputer operating system of the future and began to sink significant development dollars into it. Microsoft, however, did not appear to be as committed. In 1990, the company unveiled a new release of its Windows operating system, Windows 3.0, that overcame many MS-DOS limitations. Sales of Windows passed the one million mark only a few months after its introduction.

Questions arose about the need for OS/2. When Microsoft did little to voice its support, IBM officials became irritated and relations between the two companies began to fall apart. Relations between the two companies deteriorated. Eventually, Microsoft sold OS/2 marketing and technical development to IBM in order to concentrate on Windows.

Internetworking

Corporations needed one more piece to connect their LANs. Bridges and routers emerged as means of connecting a variety of LANs into a single wide-area network (i.e., internetworking). Initially, bridges and routers performed distinctly different functions, but recent technological advances have blurred those distinctions somewhat. Many products now offer both bridging and routing capabilities; they are called bridge/routers or brouters.

Internetworking products use local facilities, such as copper wires, or telephone services, such as AT&T's Digital Data Service (DDS) lines, to connect LANs running in a variety of locations. They have become the infrastructure that companies use to link all of their LANs.

The bridge and router market followed others and exploded. A few years ago, Cisco Systems was a small start-up company. Now, year-end revenue is on course to break the $200 million mark and Cisco seems destined to eventually become a *Fortune* 500 corporation.

As bridges and routers began to take hold, a change occurred among microcomputer users. Support issues became too complicated for many microcomputer managers to handle. While these employees were versed in using programs (such as Norton Utilities), they were not trained to manage connections to remote locations. These managers began to look for help and eventually turned to the MIS department.

Marrying PCs and Mainframes

A second trend now occurred. Corporations took stock of their information and began to understand that crucial data was stored in microcomputers on executives' desks, as well as on midrange computers and mainframes. They needed to marry all of these systems—and they turned to the MIS department for help.

As a result, the MIS manager was called in to bring order to a muddled situation caused by a 10-year period of uncontrolled microcomputer purchases. Naturally, these MIS managers did not like what they found. Employees may use two or three different vendors' spreadsheets; four or five different word processing packages; a variety of desktop operating systems; and multiple LAN technologies.

Restoring order to this confusion comes in addition to supporting traditional systems such as IBM mainframes and midrange systems. UNIX computers have now evolved in a manner similar to microcomputers and represent another factor in a complex equation.

Quickly, MIS managers discovered that basic connectivity products to connect these different devices may not be available. Management tools almost surely will be lacking in at least one area and probably in several. The task of integrating this hodgepodge may seem overwhelming, and in some corporations it is.

MIS Control

MIS managers must now take steps to make networks a bit more manageable. One approach is to choose one LAN technology and one set of applications and try to force all departments to use them. If a department is working with a different package than that specified (e.g., WordPerfect instead of Word), then users are asked to trade it in for the standard package.

Some managers have taken this approach with little or no success. In many instances, this stringent approach only leads to alienation between users and the MIS department. Unfortunately, the time to to establish central corporatewide standards was 10 years ago; MIS managers cannot turn back the clock. Users who have been purchasing and using software for years with a large degree of independence tend not to react well to tightened restrictions.

Obviously, MIS departments cannot be held responsible for every application and every possible network configuration. In most companies, the department must outline a set of standards, which include the most popular packages, LAN technologies, and standard packages used in their company. Rather than being the technical evaluator, the MIS department is evolving into an internal standards body, responsible for developing outlines so users can share information.

Taking Inventory

To set standards, the MIS department first must know what products users have purchased. Thus, the first task in LAN data management is to inventory existing equipment. To do this, MIS departments need cooperation from microcomputer and LAN users. These users may feel threatened in the worst case, or aggravated in the best case, when asked to cooperate with a central authority. MIS managers then have to choose between wielding a big stick or dangling a carrot to gain cooperation.

Of course, the company president carries the biggest stick in the firm. Because many corporations understand the value of information and realize that computers generate most of the data in their firms, many company presidents have some degree of technical savvy. If the MIS manager can convince the company president that an internal audit and central control is needed, then the company can issue a memo asking for one, and the executive vice presidents will then round up the needed data.

Many executives vie for the company president's attention, so the MIS manager will need ammunition to gain access. Some inside information may help. Computers normally represent the third largest expenditure in big companies. Only personnel costs and travel and entertainment expenses account for a larger percentage of corporate expenditures. This information may help convince the company president that tighter control of this important corporate asset is needed.

"Dangling Carrots"

If after using this information the MIS manager still cannot gain the support of top management, then more ingenious methods must be tried. One method is to enlist the cooperation of the company's microcomputer managers. Whenever a department installs a LAN, one employee, usually willing but sometimes by decree, becomes the primary support person. This employee gains an understanding of how the department uses its microcomputers and the most important support issues. Often, this person becomes an expert on many different packages because he or she often evaluates the latest software. The MIS manager should identify such employees and then gain their cooperation.

One way to do this is to facilitate the spread of information among the company microcomputer managers by forming a corporate microcomputer support group. This group can meet regularly and discuss issues, such as the best way to train new employees in WordPerfect, or exchange information about different programs, such as ways to circumvent bugs in a new release of NetWare.

Members of these corporate support groups should be chosen by the individual department heads rather than the MIS manager. Department managers are in the best position to determine the best person for the group. Therefore, it is important that the MIS manager convince the department manager that the support group is important.

Bulk discounts represent another enticement. Most departments purchase their software either from a local retailer or through catalog. In these instances, the department often pays close to full retail price. Bulk purchases can lower department purchases from 5% to 20%, and most department managers would quickly take advantage of such an offer. Bulk discounts offer MIS another advantage. If hardware and software came from a central authority, the MIS department can get a better handle on purchase trends.

Corporate Structure

The success of such efforts often depends largely on corporate structure. The level of cooperation between departments differs dramatically from corporation to corporation. In the best-run enterprise, a spirit of cooperation is fostered, so all managers work to make such groups successful. In many companies, however, departments may feel the support group is either a waste of time or a threat because departments compete against each other for funding, and in some instances even for business.

If inter-departmental cooperation is at a minimum, the MIS manager must get out the big stick, set down a set of strict rules, and limit access to corporate resources. In most instances, the MIS department is responsible for the wide area network—this often becomes the MIS manager's bargaining chip. MIS can make it mandatory that any department wishing to be connected to the WAN assign one person to the support group and complete an inventory of all equipment. If some departments refuse, MIS can simply not tie those departments into the corporate network.

Once MIS establishes a set of rules and builds a corporate infrastructure, it must complete an equipment inventory. Managers should prepare for a worst case scenario. As mentioned earlier, many microcomputers came in through the back door, so companies may not know how much equipment users have acquired during the past 10 years. MIS may find that the company probably has 50%

more computers and twice as many software programs as it anticipated. In some instances, even those estimates may be low.

Common Products

The chance that all departments are using a single set of computers, applications, and LANs is slim. But some distinct trends will emerge. In most companies, word about a particular product or package will spread from department to department and create common denominators. The MIS department must establish a priority list of devices and applications that will be supported. The department will allocate the bulk of its resources to the most popular products and packages.

Chances are good that most departments will have IBM-compatible PCs. In most instances, Apple Macintoshes represent a hit-or-miss proposition. Either half of all microcomputers users use Macintoshes, or only a small percentage of users have purchased the machines. The degree of acceptance seems to hinge on whether or not a few departments took a chance and began using the machines when they were first introduced. Once hooked, Macintosh users tend to be extremely loyal to their microcomputers.

Packages such as dBase, Excel, Lotus 1-2-3, and WordPerfect tend to be common; Ethernet and token-ring are the most popular LAN technologies. NetWare is the principal LAN operating system, but loyal IBM shops will have purchased LAN Server, and some technically savvy groups will have installed VINES.

After determining the list of approved products, the department must establish how much support to offer niche products. In most cases, the MIS department leaves application issues, such as a bad macro, to the department with the niche product and volunteers to address any network issues.

LAN Management Tools

Once MIS puts an infrastructure in place, departments need products to centrally manage a series of remote LANs. The good news is that corporations can purchase products that perform a myriad of specific tasks. The bad news is that the products are not well integrated and offer only pieces to a complex puzzle.

Management of LAN data touches on many issues including physical network management, system network management, file transfer, software distribution, and data access. Vendors only recently began try to address these different functions in a consistent manner. Products that claim to solve these issues do so only in a rudimentary manner and fall far short of customer wish-list requirements.

When LANs were still emerging, large vendors such as Digital Equipment Corp. and IBM tried to improve the capabilities of their wide area network management packages. Digital was working on an architecture called Enterprise Network Management while IBM was beefing up its NetView product. In general, such products do a mediocre job of LAN network management.

Other vendors moved in to fill the gaps. Some concentrated on physical network management, which means checking connections between different workstations. For instance, Hewlett-Packard, Network General, and Novell have delivered protocol analyzers that monitor packets traveling over LANs and determine the location of any problem. In general, these products translate hexadecimal data into understandable English (such as a break between station 4b and 5a), and can be used by either LAN administrators or MIS professionals.

LAN Management Vendors

A group of start-up firms formed to address LAN system and data management issues. Brightwork Development, Cheyenne Software, Frye Computer Systems, LAN Systems, and Preferred Software are some of the vendors that sell LAN software management packages. These vendors' products have many similarities, but differ at least slightly from one another.

Most of these products run only on Novell NetWare LANs due to NetWare's dominant market share. Some vendors are now working on versions for Banyan's VINES and Microsoft's LAN Manager. Prices for the packages range from $100 to $1,000 per server. Most of the packages are sold by value-added resellers (VARs) rather than by direct sales forces.

The different tools concentrate on various system management issues. One area is performance monitoring, which gauges the throughput of LAN servers. Application software problems is another area where the tools offer help. They allow a central administrator to access a remote user's terminal, run an application, and determine problem causes.

A third area is configuration management. The software determines what type of computers are on a network and which software programs a user is running. These features can be especially helpful to an MIS department trying to determine user configurations.

Brightwork Development

Brightwork Development (Tinton Falls, NJ) has been in the LAN network management utility software market for five years and sells five products that run on Novell NetWare LANs. During its history, Brightwork's sales have steadily risen and recently passed the $6 million.

Brightwork's *NETRemote+* lets a central administrator take control of a user's screen. The administrator can run tests to determine the origin of any network problem. *LAN Support Center* works in conjunction with NETRemote+ and stores network problem information in a central database management system. The software stores data about progress made in fixing a bad connection and produces trouble tickets that detail each step in a recovery. LAN Support Center automatically tracks support calls, generates reports, and maintains inventory databases.

Brightwork's *LAN Automatic Inventory* automatically builds an inventory of LAN software and hardware components. Device characteristics include CPU type, numeric coprocessor boards, type of micro channel, clock speeds, fixed disk drives, diskette drives, keyboard type, base memory, and extended memory. The software scans disk drives and can identify more than 1,000 software programs. Users can tailor the software to identify other software packages.

Sitelock is a virus protection and software licensing metering tool. The virus protection feature prevents users from running altered or unauthorized files. The licensing metering tool lets an administrator define the maximum number of users for an application and locks out users whenever that number is reached. Sitelock generates reports so users can see how many times a program is used and indicates when a company should purchase additional copies of a program. *PS-Print* acts as a print server and connects microcomputers to various printers.

Frye Computer Systems

Frye Computer Systems (Boston, MA) sells three products that are collectively called Frye Utilities for Networks. *NetWare Management* produces weekly maintenance reports identifying hard disk, random access memory, and other types of problems. The software automatically documents microcomputer installation options, login scripts, user names, group names, and trustee assignments. NetWare Management produces configuration generation reports, which can be used to restore a previous network configuration if a problem takes a network down.

NetWare Early Warning System automatically checks server performance at preset intervals. This package has connections to electronic mail systems running Novell's Message Handling System (MHS) standard. The Frye software can also monitor remote connections to minicomputers and mainframes, and initiate procedures with these machines.

LAN Directory features automatic updating of workstation and bridge configuration information. The data is stored in a database; warranty, service, primary support telephone numbers, and operating history can be added. Reports demonstrate inventory, audit, updating, and other management functions.

LANSystems

LANSystems (New York, NY) offers three LAN management utilities. *LANSight* integrates remote control, diagnostic, and analytical tools. An administrator can view and control any workstation on a network. The product determines hardware configurations, such as microcomputer type, CPU, video adapter, network interface card, and all types of memory as well as installed software including memory map, DOS version, NetWare shell, and memory-resident programs. LANSight requires only 4K of memory in a remote workstation.

LANSpool transforms any workstation into a print server capable of working with five printers. The product includes print selection features that pop up with any application and display menus with all available network printers. Users can send information to a printer without leaving their application.

LANSpool SI is designed to move information from a workstation to an HP LaserJet printer. Because the product was designed for the HP printer, LANSpool SI can move information to that device 20 times faster than other products, according to LANSystems. The software also notifies users when a document has been finished, the printer is out of paper, or a printer is taken off-line.

Cheyenne Software

Cheyenne Software's (Rosyln, NY) LAN management utility, *Monitrix*, keeps an inventory of network devices. With the software, a user can run a series of point-to-point connectivity tests, performance analysis, virus detection, and network monitoring. The software will also send a message to a network administrator by a pager or electronic mail.

Preferred Systems

The history of Preferred Systems illustrates the problems companies face in trying to control their LANs. In 1989, Preferred Health Care Ltd. decided to move from IBM AS/400 computers to Novell NetWare LANs. The number of LANs grew quickly and the administrative staff had trouble loading servers with standard software. The microcomputer systems manager (Jack Serfuss) wrote a program to accomplish that task. The software eased administrative chores by automating predefined configuration data and cut the task from a few weeks to a few hours.

Preferred Health is constantly looking to expand beyond its traditional customer base, so the company decided that the new package was good enough to enter the software market. In April 1990, Preferred Health Care Ltd. launched Preferred Systems and made Serfuss president of the new firm.

In January 1991, the start-up launched two products, *Origen* and *LANStandard.* Origen automatically configures new file servers and reduces the process from weeks to hours, according to the company. LANStandard provides administrators with tools to manage any Origen-configured server.

Software Distribution

Software distribution is a problem for many companies. In order to offer its users software at discounts, the MIS department must have a mechanism to move the software from location to location. Currently, the process is difficult and companies rely on manual procedures such as internal and external mail. One firm spent more than $1 million a year mailing new releases of microcomputer software to its users.

Help for managing software distribution is coming from a variety of places. File transfer products already move information from one computer to another. But these products are not suited to distributing one file to thousands of users on a complex web of LANs.

Recently, some file transfer product vendors improved the capabilities of their systems and positioned themselves as providers of software distribution programs. Spectrum Concepts (New York, NY) developed *XCom6.2*, the first file transfer product based on IBM's Logical Unit 6.2 (LU6.2) protocol. Earlier this year, Spectrum Concepts shipped a beta version of a software distribution product based on XCom6.2.

Systems Center (Reston, VA) markets *Net/Master*, the only direct competitor to IBM's NetView network management system. In addition to Net/Master, Systems Center developed *Network Data Mover (NDM)* software, a file transfer program. NDM was designed to move large files, often called bulk data, from one mainframe to a second. The product has been enhanced to more easily move information between smaller computers.

IBM provides a file transfer program called *NetView File Transfer Program*, which is also geared to bulk data transfers. In January 1991, the company unveiled a version that runs on IBM AS/400 computers, and pledged that additional versions would follow. In September 1991, IBM unveiled a second distribution product, *SAA Delivery Manager*. This product, which fits under the AD/Cycle umbrella, was designed to electronically move programs, text, and documentation from mainframes to workstations.

Travelers Insurance Synchrony

Because there are so many missing pieces, large companies are building their own data management packages. For instance, Travelers Insurance, which has emerged as a technology leader among IBM customers, needed a tool to distribute software to microcomputer users. Rather than wait for a vendor solution, the company wrote its own package. *Synchrony* downloads software from IBM mainframes to microcomputers or LAN workstations.

Synchrony runs on microcomputers with Microsoft's MS-DOS or OS/2 operating systems. The software works with IBM's LU6.2 communications protocol so that workstations and mainframes can share information. A data compression feature insures maximum use of available bandwidth. From a central console, a network technician can retrieve files, generate reports, and monitor software information. A checkpoint restart feature automatically restarts any problem transfer.

Travelers Insurance began designing Synchrony in 1987. A beta test version was installed in May 1988 and a production version followed in the fall of 1988. Synchrony operates on 8,000 of the firm's 20,000 microcomputers. Although it saw that Synchronoy had market potential, the insurance company decided not to become a software supplier. Instead, Travelers Insurance sold the marketing rights to Packet/PC (Farmington, CT), a supplier of communications software. In September 1990, Packet/PC announced commercial availability of Synchrony. (Packet/PC is now known as Telepartner International.)

Change Management

Distributing software solves only one portion of the client/server software management problem. Software often changes; thus, corporations need tools to ensure that all users are working with the most recent version of a program. Change management programs have been available for years on mainframes. On a mainframe, software changes can be tightly controlled because software resides only in the mainframe. MIS departments can establish procedures so only selected users, primarily programmers, can change programs.

In a client/server environment, controls are not as tight. Software resides on thousands of workstations as well as the mainframe. Almost any user has the authority to alter at least some portion of a program.

Recently, change management program suppliers began to grapple with client/server software issues. In January 1991, Legent (Reston, VA) announced *Endeavor*, which includes mainframe and microcomputer change management software. The microcomputer portion works with programs developed on microcomputers running Microsoft's MS-DOS or OS/2 operating systems and the mainframe portion runs on IBM mainframes with IBM's MVS operating system.

Bank of America CMF

In 1987, Bank of America decided to write its own management distribution program. The software, dubbed *Change Management Facility (CMF)*, was initially used in a pilot program to automate bank branch offices. As part of the pilot program, the bank installed IBM microcomputers and token-ring LANs at branch offices, and designed software that linked the remote microcomputers to IBM mainframes.

CMF ensured that branch office users were working with up-to-date software, and helped central network technicians manage remote connections. CMF now runs on approximately 1,100 LANs and supports more than 15,000 microcomputers at 850 Bank of America offices.

CMF's Central Distribution Facility module runs as a CICS application on an IBM MVS mainframe. A network operator can identify microcomputer hardware configuration information to determine if there is sufficient memory for a new release of software. A central operator can also define distribution and installation schedules, and authorize data transmissions.

A Distributed Management Facility feature executes file transmissions. The software suspends transmission if a network problem occurs and automatically resumes the transfer once the problem is solved. A screen display illustrates distribution status and indicates any problem.

Bank of American also had the option of selling its software to other users. Instead, the bank opted to sell the distribution rights to IBM. In the fall of 1990, IBM began marketing the product as *Distributed Change Management Facility (DCMF)*.

Developing such products is not trivial; Bank of American estimated that its expenses surpassed $10 million. So, it is unlikely that most users will take on the task. But even the Bank of America product falls short of some user requirements. DCMF runs only on Microsoft's MS-DOS operating system, although a version for OS/2 is under development. Also, DCMF works only with IBM's LAN Manager and PC LAN Program network operating systems; a version for Novell NetWare LANs is also being designed.

Data Sharing Standards

These tools can help an MIS department manage data, but they do not deal with the fundamental issue of how users can access information. Corporations want to tie microcomputers to LANs so users can share information stored on different systems. Unfortunately, tools to perform such tasks are only in the nascent stages.

The biggest problems are the wide variety of data storage methods, and the locations where it is stored. The half-dozen or so major application categories store data in slightly different manners, which can limit users' access. A few tools have evolved to help solve this problem. File systems, such as Sun Microsystems Network File System (NFS), outline consistent standards but only a few microcomputer applications use them. In general, NFS is used more on UNIX workstations.

DBMSs were designed as central data storage areas. Vendors tout distributed databases as the solution to these problems. But distributed databases are not yet mature enough to be widely implemented.

Accessing DBMS Data

Many DBMS products rely on proprietary technology, but proprietary walls are slowly being broken down. Structured Query Language (SQL) is emerging as a foundation for data sharing. However, vendors are moving in several directions to overcome SQL limitations and the chances of a single set of standards appear dim.

Two years ago, a group of vendors including Digital Equipment Corp., Hewlett-Packard, and Oracle formed the SQL Access Group. The consortium's goal was to outline a set of standards so that information and applications generated on one DBMS would be able to run on another DBMS. In April 1991, the group completed its standard; and in July 1991, vendors in the group demonstrated prototype implementations of SQL Access Group compliant products. Delivery of these products is expected to begin in 1992.

The SQL Access Group has been criticized because its standards rely on the Open Systems Interconnection (OSI) reference model. OSI has been touted as a way to connect different vendors' networking equipment, but has gained little market acceptance to this point. Instead, users turned to the Transmission Control Protocol/Internet Protocol (TCP/IP) to link different systems. Also, the SQL Access Group standards leave room for interpretation. Vendors could deliver compliant-products that do not work together and do not offer users any additional functionality.

Alternatives Arise

In September 1991, IBM (which is not a member of the SQL Access Group) unveiled the Information Warehouse, a set of standards that compete with the consortium's work. The IBM standards include a set of APIs that have been or will be incorporated into IBM DBMSs, such as DB2 and OS/2 Extended Edition Data Manager. Companies such as Computer Associates International and Oracle have pledged to support the IBM Information Warehouse. Unfortunately, many Information Warehouse components are not scheduled to be delivered until Fall 1992. Also, the IBM model relies on the company's LU6.2 protocol, which, like OSI, has not gained as much industry acceptance as initially expected.

Microcomputer vendors are advocating their own technologies as industry standards. For instance, Apple is pushing its Data Manager and Data Access Language as ways to provide transparent data access. Computer Associates and Tandem Computer have delivered products that provide Apple Macintosh users with transparent data access to their DBMS. At this point, Apple probably does not have sufficient market share to transform its products into de facto standards for data access.

In November 1991, Microsoft unveiled Open Database Connectivity, an API that application developers can use to provide transparent data access. The Microsoft standard is based on work done by the SQL Access Group. The Microsoft specifications are viewed as appropriate only for the firm's Windows operating system and may not be suited to complex collections of data.

Summary

Currently, all of the data integration standards represent only a first step to widespread data integration. Flushing out these standards will take several years and thus offer little short-term help to MIS managers in need of an immediate way to improve connectivity.

Because of current limitations, many companies are establishing procedures to restrict the movement of data between LANs and larger systems. In most instances, corporate data continues to be housed on mainframes. Users can download information from a central system, work with it on a microcomputer, and then send it back to a central system either when the task is completed or at the end of the day. Very little production data is stored and kept on microcomputers, and that will probably remain the case for the next few years.

No one is certain what will happen during the next few years, but MIS departments are being asked to insure that their companies will be in a position to take advantage of technical advances. MIS departments are being asked to rely on future standards from suppliers who have changed their minds constantly over the years.

MIS departments should have no illusions about vendor commitments. They will be forgiven if they are a bit skeptical. The industry is changing rapidly and vendors are moving in conflicting directions. Strategic products are being discarded before they have been delivered and vendors are teaming up and breaking up like a confused group of teenagers.

The industry's largest company has sent out confused signals. Three years ago, IBM was touting its LAN Server product as the LAN operating system of the future. Early in 1991, the company signed a joint-development agreement with Novell and now plans to integrate NetWare support into many IBM products. IBM pushed its OS/2 operating system as the foundation for next generation client/server applications. In the fall of 1991, the company announced a joint-development plan with Apple for a new object-oriented operating system. How these two items will play together is not clear, probably not even to IBM.

As a result, MIS managers have to be extremely cautious and outline standards that are subject to major revisions. In this scenario, it becomes increasingly more important for them to take on the role of peacemaker than that of technology leader. If they can convince department managers of the benefits of cooperation, then their company should be ready for next generation computing—whatever that might entail. If they continue to view themselves as technical gurus, they and their corporations may be in for problems. The computing world is very different from what it was 10 years ago, and shows no sign of stabilizing. ■

This report was written exclusively for Datapro by Paul Korzeniowski. Mr. Korzeniowski is a freelance writer based in Acton, MA. He has covered the computer industry for nine years and has extensive knowledge about mainframes, wide-area networking, microcomputers, and software.

Establishing Data Administration Strategy

In this report:

This report will help you to:

- Manage the functions of data administration as a business within your organization.
- Identify the market needs in your organization which data administration products and services can help satisfy.
- Add value to the services and functions provided by data administration in ways that are perceptible to the rest of the organization.

Data Administration as a Business

Some time ago I was asked to speak on the value added to an organization by the data administration function. I said sure, then scrambled to put together some material. The way to address the topic turned out to be fairly simple: Treat data administration's function like a business.

In the first part of this report, I will discuss data administration (DA) as a business within the host organization and address some basic information management and information architecture topics. We will take a look at the markets for DA services and examine the needs of these markets. In the second part of the report, I will show how to choose the right specification languages, tools, and people.

DA is a Business

Whenever I begin to work with a new business area on information planning or modeling and that area has not already prepared an initial functional breakdown, I try on the following five high-level business functions:

- Monitor market needs and competition.
- Manage products.
- Sell products.
- Manage the relationships among parties.
- Process the work.

This Datapro report is a reprint of "Data Administration as a Business" by Thomas A. Bruce, pp. 44-56, from *Database Programming & Design*, Volume 2, Number 11, November 1989. Copyright © 1989 by Miller Freeman Publications, and "Data Administration: Making the Right Choices" by Thomas A. Bruce, pp. 50-56, from *Database Programming & Design*, Volume 3, Number 1, January 1990. Copyright © 1989 by Miller Freeman Publications. Reprinted with permission.

Figure 1.
Information is "Data in a Context"

Data Becomes Information

The following is "Data"—by itself, we have no idea what it means.

```
1 2 3 4 5
```

We can add more, but we still haven't a clue.

```
1 2 3 4 5
5 4 3 2 1
```

Once we add a definitional context, we begin to get information. Now, at least, we know there are four valid ZIP codes in the world (whatever "ZIP codes" are).

```
ZIP CODES
1 2 3 4 5
1 5 6 0 1
5 4 3 2 1
9 4 1 3 7
9 4 7 1 0
```

But it's still basically just data. If we relate this set of data to some other data, we get more information.

ZIP CODE	CITY	STATE
1 2 3 4 5	Schenectady	New York
1 5 6 0 1	Greensburg	Pennsylvania
5 4 3 2 1	Green Bay	Wisconsin
9 4 1 3 7	San Francisco	California
9 4 7 1 0	Berkeley	California

These usually provide a good starting point for discussion. Dataflow diagram folks would say the functions are unbalanced, but that's not the point.

These functions fit the DA business. *Markets and competition* are those areas both inside and outside our organizations that might benefit from what we have to offer or might offer something that addresses the same needs we are addressing. *Products* are the services provided by DA and the tools it may develop and provide to the organization. *Relationships among parties* are the alliances we must build and maintain both inside and outside the organization. *Process the work* means to do everything we have decided to do to support the business we have chosen to be in.

I will summarize a view of the value added by DA that is derived from this model. Nothing presented here is radically new; my hope is that the discussion will provide a context in which to look at some new (or remembered) insights.

Markets and Products

We must first ask ourselves one question: What are our markets and what are their needs? We can then begin to identify products that address those needs. Table 1 displays a starting set of markets. The set of markets will look different in different organizations—each organization has its own internal structure—but this will do for a start.

Table 1. A Starting Set of Markets

Markets	Needs	Products
Business clients (users)	??	??
Application development	??	??
Database administration	??	??
Data processing operations	??	??
Systems programming	??	??
Telecommunications services	??	??
Audit/EDP audit	??	??
Shareholders	??	??

Next, we must put together a set of needs for each market and ask whether these needs currently exist and whether they are recognized as needs. If not, do we have to create them (or at least raise the awareness level)? In many organizations the data folks know the needs are there, but sometimes the business folks don't.

I have seen a lot of frustration around DA groups. It generally seems to result from situations where "long term is sold short for the short term," or "no one can get the point across that data is important." This means either that the organization does not need DA (unlikely) or that DA has not addressed its function from a basis that will allow it to recognize where it can and must contribute. Either way, there is some educating to do before awareness of the need for DA rises to a level where the organization chooses to act.

Education is necessary before value can be perceived. It leads to awareness, and that leads to demands for the satisfaction of needs. Providing this education falls under the "Sell Products" function of our DA business.

Structure for Education

One way to approach education in the information area is to discuss the DA function within the context of an information management environment, which in turn is part of a larger framework used to manage an organization's systems resources. Once information management concepts are established, it becomes possible to discover where DA can best contribute. What follows is a brief outline for this education process. We will then return to discussing markets and needs.

I often use several questions as a guide: What is information? What is information management? What is an information systems architecture (ISA)? What is information planning? The list may seem overly simplistic, but it has proved surprisingly useful.

To start, I would claim that information is "data in a context" and use the example shown in Figure 1 to get the idea across. (It's an old one; as a gentleman who attended the presentation pointed out, it works equally well starting with two lights on a black background; adding an overlay of the Old North Church, the context finally becomes clear.)[1]

From this example, we can claim that to get information we need the definition of the "data" stuff (a ZIP code is a number assigned to a discrete geographic area by the U.S. Postal Service to facilitate delivery of the mail) and the definition of the "structure" stuff (the ZIP code 12345 is for a city named Schenectady, which is in a state called New York).

So to get the information we need to capture the definition of data and the structure of the context. It's fun to extend these thoughts—what contexts transform information into knowledge and knowledge into wisdom?—but this isn't the time.

Thus, information management could be characterized as the practice of giving context to data and doing so in a manner that employs appropriate business management practices and techniques. "Giving context" consists of developing and using standards for:

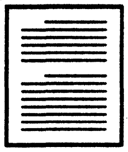
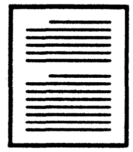
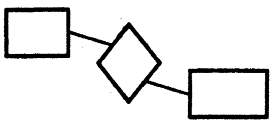
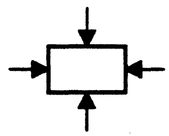
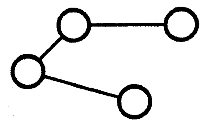
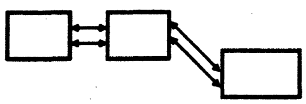
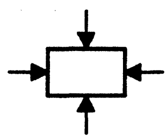
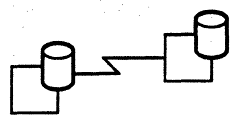
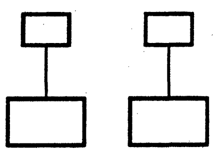
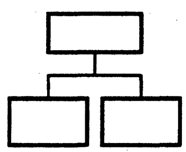
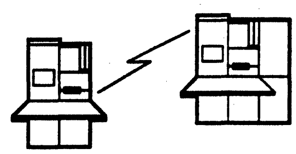
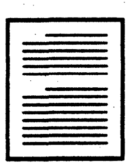
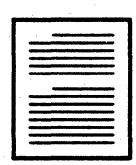
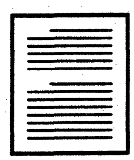
- Information planning: determining what data and structures are important to analyze and when.
- Data definition: specifying what data objects need definition and the components of these definitions.
- Data structure: deciding upon the right structural information to capture, when to do it within the project life cycle, and how (choosing the modeling techniques and tools).
- Metadata access: determining how to use a repository to hold the information and defining mechanisms to control access to this information.
- Context presentation: deciding how and when to use appropriate presentation techniques to make the context evident.

Information management is important in every stage of business planning and systems development. Some organizations refer to it as a data management function. It is also referred to as information resource management.

ISA

Architecture has taken on so many meanings that its use contributes to an awful lot of content-free discussions. It has a sky-high fog index. One of the most overused buzzwords of the 1980s, it means so many things to so many people that it often retains little value other than to help communicate such

Figure 2.
A Framework for Information Systems Architecture

	Data	Function	Network
Objectives/ Scope	List of Things Important to the Business  Entity = Class of Business Thing	List of Processes the Business Performs  Process = Class of Business Process	List of Locations in Which the Business Operates  Node = Business Location
Model of the Business	e.g., "Ent/Rel Diag"  Ent = Business Entity Rein = Business Rule	e.g., "Funct Flow Diag"  Proc = Bus Process I/O = Bus Resources (Including Info)	e.g., Logistics Network  Node = Business Unit Link = Business Relationship (Org, Product, Info)
Model of the Information System	e.g., "Data Model"  Ent = Data Entity Rein = Data Rein	e.g., "Data Flow Diag"  Proc = Application Function I/O = User Views (Set of Data Elements)	e.g., Distributed Sys. Arch.  Node = I/S Function (Processors, Storage, etc.) Link = Line, Char.
Technology Model	e.g., Data Design  Ent = Segment/Row Rein = Pointer/Key	e.g., "Structure Chart"  Proc = Computer Function I/O = Screen/Device Formats	e.g., System Arch  Node = Hardware/Sys Software Link = Line Specifications
Detailed Representations	e.g., Data Design Description  Ent = Fields Rein = Addresses	e.g., "Program"  Proc = Language Stmts I/O = Control Blocks	e.g., Network Architecture  Node = Addresses Link = Protocols
Functioning System	**e.g., Data**	**e.g., Function**	**e.g., Communications**

broad concepts as *strategic*, *structured*, *flexible*, and other noble things. Generally, it means some unifying structure.

It might be better to just strike *architecture* from the vocabulary than attempt to describe how it fits the systems development business. However, since everyone will use it, it seems best to adopt a framework in which the term can take on some useful meaning.

John Zachman of IBM has provided what I feel is a very useful framework for discussing the architecture of an information system.[2] (Bless you, John—you have given back some meaning to the word *architecture.*) Drawing on analogies from the construction business, Zachman's (ISA) framework describes different views of a planned system. These views are appropriate to the owners, designers, builders, and operators of the system and focus on the dimensions of data, function, and network (Figure 2). The framework can be extended to describe cross-cell views and, as Zachman puts it, "the who, the when, and the why, as well as the what, the how, and the where."

Owners' views generally concentrate on overall strategic and tactical considerations of the organization. These are wide in scope and not necessarily defined precisely (Zachman would call these views of "big fuzzy things"). Initial business planning and analysis activities will usually result in the first level of these views.

Designers' views add precision but remain independent of the technology eventually employed. Structured analysis and design, information modeling, and perhaps some level of prototyping are techniques used to develop these views. It has been my experience that things work best when the function, data, and network components of these high-level views are developed at the same time by a team consisting of business, application development, network, and DA folks. Each area brings skills and knowledge that, when combined, yield a good overall picture. If these individuals can be involved in preparing the owners' views, so much the better.

Physical implementation of logical structures is highly dependent on the characteristics of the hardware and software base chosen for the system. While the desired logical relationships are not influenced by physical constraints, realizable relationships normally are. I believe it is necessary to first ask "what do we want?" and answer that before assuming something is impossible (it usually isn't; just very difficult). Technology constrains solutions, not requirements.

It is at the *builders' views* level that technology considerations come into play. The top two levels (owners and designers) generally represent statements of business requirements. The lower two levels (builders and operators) represent statements of solutions, which are constrained by technology, time, and cost.

In the physical world, standards might specify which hardware and software will be used at various geographic locations. Depending on the characteristics of that hardware and software, they may also specify standard structures for function and data within the technology's capabilities. I choose not to call these standards views or architectures. Rather, I consider them to be templates for building architected systems. I do not mean to minimize their importance (no architecture is useful without implementation standards); I simply want to avoid the fog. The standards constrain the lower-level views.

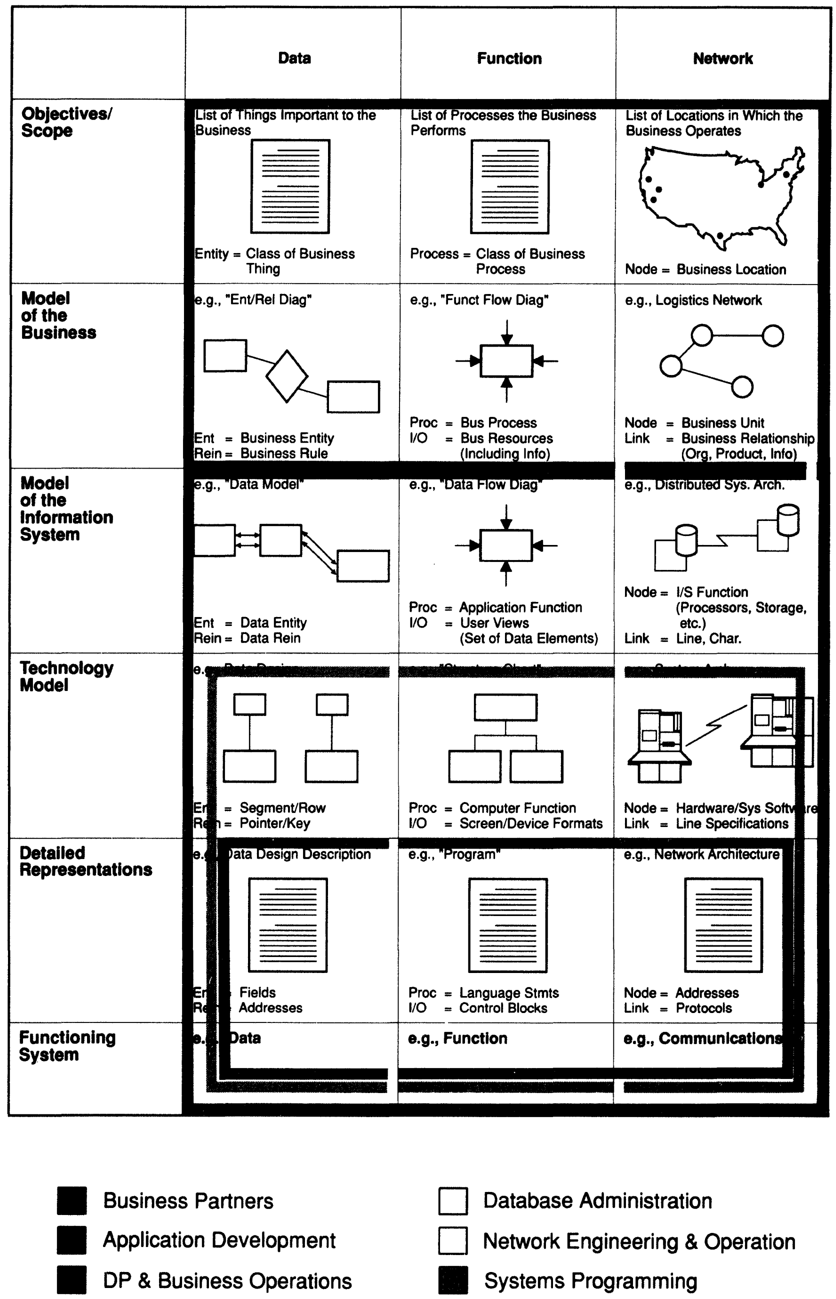
Operators' views are used for day-to-day system operation and monitoring. There are two types of operators: computer operations folks, who need run-time documentation and exception procedures; and business operations folks, who are "processing the work." (Aha! Some needs!)

To use the Zachman framework, I first need to extend it a bit and introduce the terms *architected system, architecture product, architecture process,* and *architecture practice.* Some definitions:

Those who design and build architected systems are the architecture practitioners; that is, the architects. Architected systems are those that adhere to standards set forth in the architecture products and which are developed according to the architecture processes. Architecture products are the views represented in the Zachman framework, supplemented with cross-cell and transformation documentation and any standards or templates that emerge.

Architecture processes are the means by which architecture products are developed and a consensus among all parties is reached. They are part of the architecture practice. No independent architecture group can develop products or mandate their use. Such a group will not know what the products' users need; the products' recipients will

Figure 3.
Markets, or Multiple Views, of ISA



resist using them if they are not involved in their creation. Nothing new here; most anything developed without participation and buy-in by the client or user will fail.

The processes, then, are those that allow architects to define the architecture and reach a consensus about it. These architects are the system's designers and builders, not the members of some architecture unit. An architecture unit's staff can only teach the practice, give structure to and facilitate the processes, and maintain the products.

Agreed-upon methodologies (architecture processes) are needed to ensure some level of consistency across the development organization, although these methodologies must not be so rigid as to become burdensome or overly constraining. One might consider them more on the order of guidelines than standards (although I am sure quite a debate could be had regarding the distinction). A set of methodologies applicable to different situations is needed. Methodology types include a system development life cycle, a business function analysis process, a functional decomposition and recording technique, and information modeling methods (top-down and bottom-up).

Tools (automated and manual) can speed development as well as manage and control the process. A complete set of such highly integrated and interdependent tools is needed, although I am not aware of any such set today. CASE tools are available, but, at least in the short term, the challenge is to find individual tools or sets of tools that work well together. At minimum, along with traditional tools such as editors, compilers, and so forth, we will need a functional decomposition and analysis tool, an information modeling tool, a reverse-engineering tool (for both function and data), a data dictionary/repository, and a project management tool.

Design standards (or guidelines) are needed to ensure consistency of system solutions. The architecture products and process may, over time, provide many of these.

To ensure consistency and integration of designs within a chosen set of technologies (the real, physical world), a set of technology models and design templates is highly desirable. It is difficult, if not impossible, to develop anything but an initial version of these templates in advance of development of the first architected system.

Templates emerge as logical and physical (external and internal) design of a system proceeds. They come from the grass roots as designers of systems share their solutions with others. At some point, use of templates may be mandated, but to do so from the start will diminish their usefulness and probably sink the effort. System developers don't like to be told "from above" how to design their systems; they do, however, listen to their peers and reuse ideas they find valuable. There does come a time, of course, when preventing reinvention of the wheel is desirable.

All of these architecture whatzits are products that satisfy one or more of an organization's needs. Using this ISA context and a potential list of DA markets, it is interesting to see how they correspond. Figure 3 paints six of the markets into the Zachman framework. Multiple views are needed even within a cell because there can be many viewers of that same cell information.

Mapping the architecture processes into the framework is also helpful. (For example, the overall development methodology covers the entire framework; business systems planning methods cover the top levels.) But I will leave that to you. Figure 4 contains some starting lists that can be replaced with lists that are more appropriate for your organization.

Figure 4.
Lists for Mapping Architecture Process to Framework

Architecture Products
- Preliminary requirements
- Initial system solutions
- Detailed requirements
- Functional specifications
- Hardware/software specifications
- Application software/JCL
- Area information models
- Product information models
- Project information models
- Physical database specifications
- Design templates

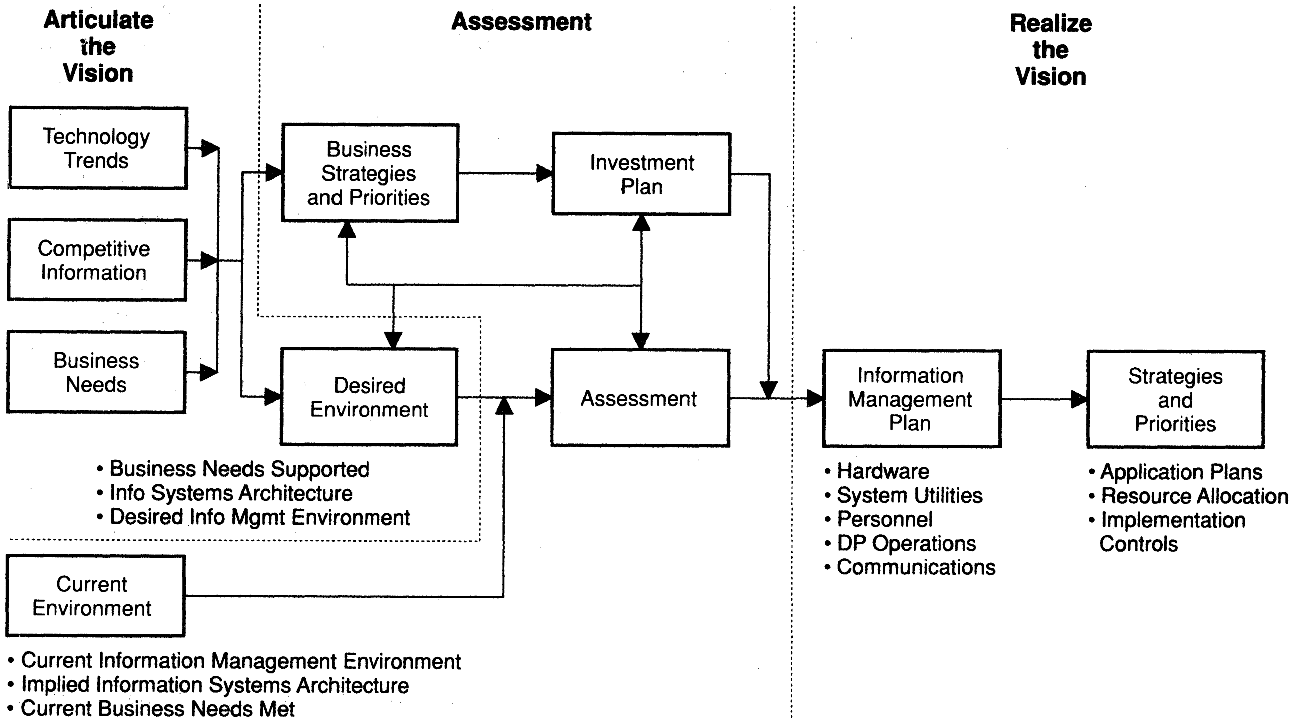
Architecture Processes
- System development life cycle
- Business systems planning
- Functional analysis
- Information modeling
- Structured design
- Structured programming
- Prototyping
- Project planning and control

Architecture Tools
- Activity/function modeling
- Information modeling
- Data dictionary/repository
- Programming
- Prototyping
- Source code library maintenance
- Project planning and control
- Reverse engineering
- Reengineering
- Editors
- Compilers

Information Planning

We can think of information planning as a front-end business activity that precedes the development of any system. It is a process that helps a business understand how to manage its information resources. Again, nothing new here; what follows may sound a lot like the old concept of "determine where you are, determine where you want to go, and develop a plan to get there." Still, it sometimes helps to lay it out. I typically use the diagram shown in Figure 5 to present these

Figure 5.
Three Phases of Information Planning



concepts.[3] I find it amazing how well it is received by business and systems managers.

The process begins with an articulation of the vision of those who run the business and a description of an environment that will allow that vision to be realized. This desired environment is described in terms of the business needs to be satisfied, the ISA required to satisfy those needs, and the information management environment needed. Many factors influence that vision, including the current needs of the organization (and its customers), trends in technology (an enabling factor in many cases), and actions of the organization's competitors.

Moving to the assessment phase demands a description of what currently exists. The desired environment description may be developed top-down, whereas the current environment description may need to be developed bottom-up—that is, determine what the environment looks like, infer what ISA supports it, and then guess what needs are being satisfied (and what needs the current ISA is intended to satisfy).

The assessment phase compares the current environment against the desired environment to develop plans and budgets for the next cycle of change. The organization's priorities and strategies enter here; possible actions are limited by the organization's investment plan. The direction for future efforts is set here. It may be, for example, that the current ISA and information management environment are adequate to support the required functionality, but business needs are not well supported. Or perhaps current needs are well supported, but the existing ISA and information management environment severely limit future support. The assessment takes place from both a business perspective (what does the business need to make money?) and a systems perspective (what is needed to help the business make that money?).

Realizing the vision is the third phase. The information management plan and a set of strategies and priorities are developed based on the results of the assessment phase. The plan describes hardware, software, human resources, communications mechanisms, and the operating environment to be constructed. Strategies and priorities are established that will yield a development plan for applications and result in well-managed allocation of resources.

And then the cycle begins again.

One might ask whether DA would normally get involved in these things. In the past, the answer would probably have been no. But this is an area where DA can contribute quite effectively. Remember, we are looking for needs we can satisfy; we are not walking around with a bag of preformed solutions in search of an appropriate problem.

To summarize, an information management environment is one in which the ISA is driven by business information needs. We have also seen that in an architecture environment, system development is driven by a set of architecture products and practices that integrate the differing views of the function, data, and network needed to define, build, and operate the system.

But the question remains: What is the role of DA in all this? To understand DA and how it can add value to the business, the context must be clear. In general, we can claim that to understand any business function—and DA is such a function—we must understand the environment or context in which that function operates.

Markets and Needs

Now that we have reviewed a framework for discussing information systems, it is time to return to our DA markets and identify the needs of each market. Here again are some starting lists; throw them out and build some suitable to your organization.

Business Clients. Our clients often need assistance in planning for future developments and identifying requirements that can be forwarded to an internal system development organization or outside vendor. They must also consider the current state of the business and systems (what is often called situation analysis) and identify future alternatives. They need to document all of this as well.

Business clients also need assistance reconciling data coming from a diverse set of current systems and assistance extracting meaningful business information from that data. Given the nature of the business world, they always need this yesterday.

It is important to remember who provides the funds to operate the DA function. You may be surprised at how well your efforts are accepted once the business understands that you are willing to listen to what it needs.

Application Development. Application developers are often called upon to help the business determine requirements and strategies. They must design function and data structures that address requirements, generate data layouts and control blocks for inclusion in programs, manage change in their applications, and create appropriate documentation. They must test everything. In many

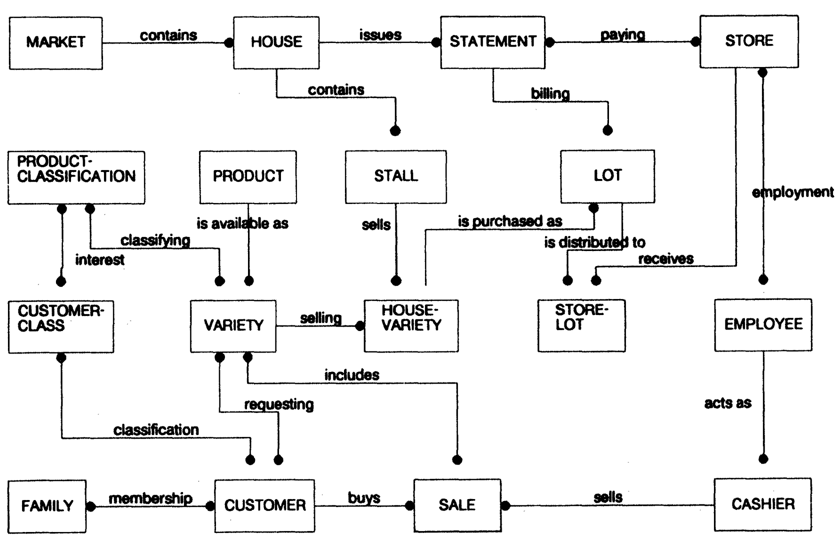
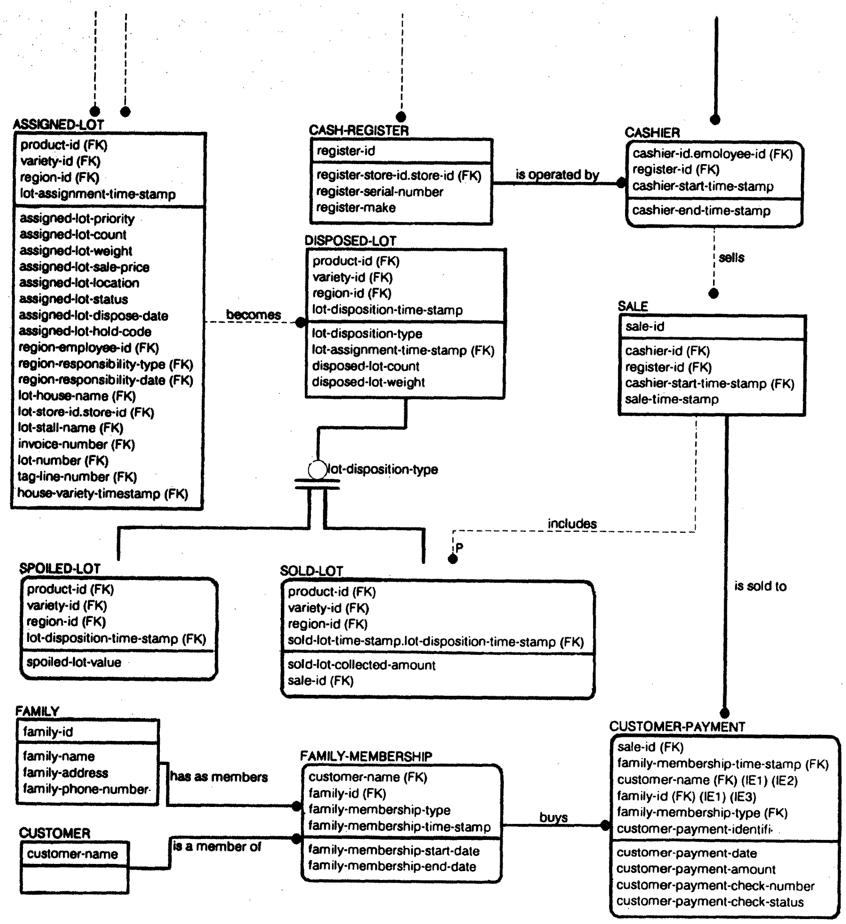


Figure 6.
Produce Market Entity-Relationship Diagram

Figure 7.
Fully Attributed Model—Lot Disposition View



cases (the majority, it seems), they must analyze old systems and bring documentation up to date. They must understand the general capabilities and specific details of packages being considered for use in the area and interface new packages to existing environments. They also need to accomplish all of this yesterday.

Education in new and emerging technologies and techniques is needed, as are refreshers on established methods. How often have you found that your application development partners have forgotten (or never knew in the first place) about such established methods as structured analysis and design and structured programming? The answer is probably "all too often."

Database Administration. DBAs need a great deal of information to design long-lasting databases. All objects in the business and database models require precise definitions. Data volumes, transaction response times, and access patterns need to be specified, and the business model needs to be transformed into a technology model of the database. Databases need to be "genned" and versions adequately controlled. And the DBA needs to have accomplished all of this, it seems, before anyone has any idea what the business requirements are. (I'm just kidding but then again . . . .)

Data administrators sometimes forget that one of the primary users of the information models it champions is the DBA. If we expect our glorious models to have some effect on the business, we had best not forget that they need to be useful to several audiences. Have you ever seen a situation in which a model was provided to a DBA and was promptly ignored? Never happens. When it does happen it is just because the DBA is stupid, right?

Data Processing Operations. The folks in data processing operations, who keep the machines running day after day, rely to a large extent on current operational and exception processing documentation. When something goes wrong, they frequently have a devil of a time figuring out how to prevent the

Table 2. DA Opportunities

	SDLC Phase					
Value Area	**1**	**2**	**3**	**4**	**5**	**6**
Business Planning Support	X	X	X	X	X	X
Business Requirements Analysis	X	X	X	X		
Database Requirements Definition		X	X	X		
Requirements Documentation	X	X	X	X		
Data Object Standardization	X	X	X	X	X	X
Logical Design	X	X	X			
Physical Design		X	X	X	X	X
Database Generation				X	X	X
Copy-Code Generation				X	X	X
Testing Support			X	X	X	X
System Documentation	X	X	X	X	X	X
Inventory/Change Management	X	X	X	X	X	X

SDLC Phases:

1. Business Planning
2. Requirements Analysis
3. Logical Design
4. Physical Design
5. Coding/Testing/Installation
6. Operation/Maintenance

problem from spreading. They need to solve production problems quickly. They also need to assess the impact of pending changes on the processing environment and control these changes in an orderly manner. Here, possibly, is a real opportunity.

Systems Programming. The folks who install new machines and new releases of operating systems and shop utilities also need good documentation and access to an inventory of some sort to manage change. Systems programmers may also benefit from information about what software releases are operating where, what licenses exist, and so on. Standards for the use of general-purpose utilities and the general structure of jobs that will run in the machines can also be most helpful.

Telecommunications. The telecommunications folks need to manage networks and devices, know what is running where, get broken things fixed quickly, and introduce change in an orderly and controlled way. Once again, standards, good documentation, and good configuration management and change control information will be helpful.

Auditing. Auditors (both those concerned with business audits and those concerned with EDP systems and operations) need to be able to identify whether systems are operating correctly and whether they conform to specific organizational policy and generally accepted accounting practices. To do this, they need documentation of functions and data and the requirements the systems are intended to address, as well as capabilities for retrieving, summarizing, and reconciling information. In your organization, when was the last time anyone bothered to ask the auditors what they needed?

Shareholders. How about your shareholders? Do they care if you have an enterprise model or the world's largest communication network? Not usually. I own a small amount of stock in several companies, and all I care about is that the company be profitable and grow (so that I see some return on my investment). I happen to believe that strategic systems planning, integrated databases, and so on will, in time, result in higher levels of profitability and growth. Therefore, I tend to consider these things when choosing where to invest. But don't try to convince a shareholder (or senior manager, for that matter) of this unless you can show precisely

Table 3. Critical Points

Value Area	SDLC Phase: 1 2 3 4 5 6
Business Planning Support	E——K-• •-A—>
Business Requirements Analysis	E——K-• •-A—>
Database Requirements Definition	K-• •-A——X->
Requirements Documentation	E——K-• •-A——X->
Data Object Standardization	E——K-• •-A——X——X——D->
Logical Design	E——K-• •-A—>
Physical Design	• A——X——X——D->
Database Generation	• X——D——D->
Copy-Code Generation	• X——D->
Testing Support	• A——X——X——D->
System Documentation	K---A——X——X——D->
Inventory/Change Management	E——K——A——X——X——D->

Information Model Level:
E: Entity-Relationship Diagram
K: Key-Based Model
F: Fully Attributed Model
X: Transformation Model
D: DBMS Model

how. That's tough when you are discussing the subject with someone whose sights are leveled on next quarter's results.

Well, it has been a long introduction, but we are finally ready to discuss DA and the products it can offer. These products carry the value added by DA. That value exists only if it is perceived.

Making the Right Choices

As a colleague once reminded me, if you enter a maze and make the wrong decision at the first turn, it doesn't matter how good your decisions are from that point on.

In this report we've consistently returned to the theme of conducting the data administration function as a business within the host organization. Like any such business, DA must focus on the needs of its markets. The opportunities to be addressed should be chosen based on business principles, not technical ones.

DA must strike a balance between considering current needs and addressing anticipated needs. The needs and opportunities will change over time as the organization begins to understand the value DA can add. Address current needs now, lay the foundation for addressing future needs, and always pay attention to politics.

As professional data administrators, we must be prepared to push our ideas and fight for them—but we must also be prepared to back off. If the choices the organization makes seem irrational, take another look at its needs. If two people come to different conclusions based on the same facts, the values being used to make the choices must be different. We must be prepared to change the value system of the organization, but we must also accept that this will take time and be prepared to work within the existing value framework.

Timing is Everything

Would you give a diagram like Figure 6 to a DBA as the basis for physical database design? Would you show a diagram like Figure 7 to a high-level business manager as an aid in discussing general business strategy? I hope not.

When we think about the tools of the data administrator, such things as data dictionaries and CASE come to mind. Those tools are needed, but it's important to remember that the methods and techniques the data administrator uses are as much tools of the trade as the software and hardware.

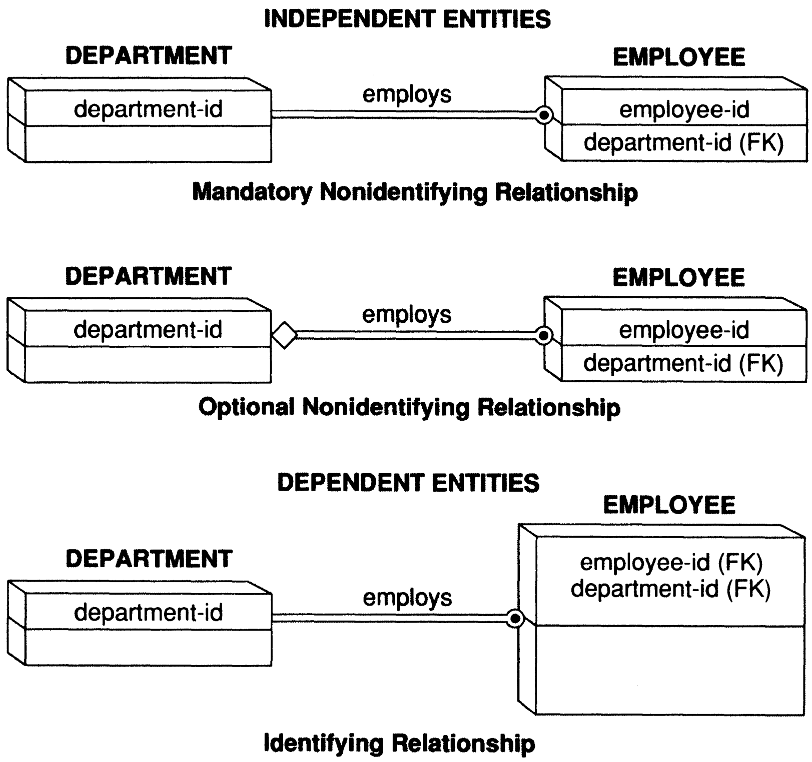


Figure 8.
IDEF1X Entities and Relationships

Just as we must avoid presenting the wrong diagrams to the wrong audiences, we must be careful to match the right tools and methods to the right tasks. The diagrams shown in Figures 6 and 7 are only valuable if we use them in the right places.

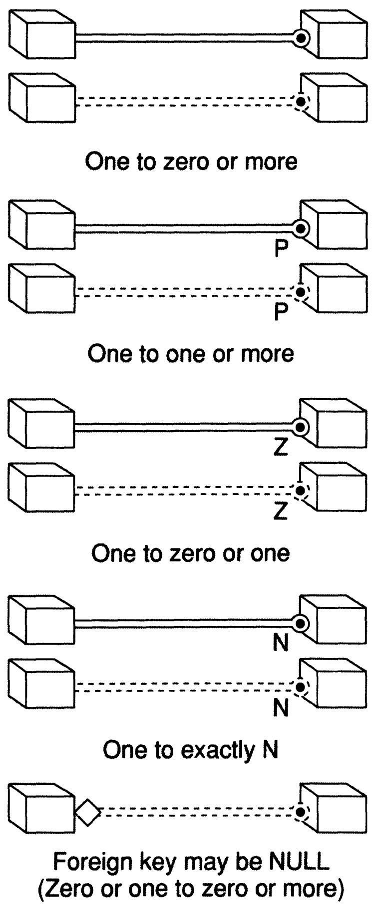
Table 2 shows DA opportunities from the perspective of the classical system development life cycle simplified into six phases. The table is the result of a matrix that matched the markets for DA services with their needs, identifying places where DA could help. The DA opportunities are indicated by an X. We use various methods and tools to address these opportunities—and sometimes invite failure by applying the wrong method to the wrong opportunity. So before we look at the more traditional tools of the data administrator, let's look at one of the methods: information modeling.

Table 3 shows where building various levels of models can be useful for a top-down development project. The model levels listed here are taken from the IDEF1X discipline (extended a bit) and generally correspond to the Zachman levels of scope and objectives, model of the business, model of the information system, technology model, and operating system. The higher-level models support the early project phases, while the more detailed models support the later phases.

Each line of the table shows two critical points. An arrow (>) shows a point beyond which further modeling provides no additional value. An asterisk indicates a point at which attempting to provide information-modeling services for the project will most likely be perceived as disruptive if this is where modeling activities start. For example, assisting the requirements documentation activity with a fully attributed model during phase 3 (logical design) has the most value if the higher-level models were prepared during earlier project phases. Starting the modeling activities during phase 3 is extremely difficult without the higher-level models and is likely to slow the project down. This delay is not likely to be perceived as adding value.

When to start and when to become involved with an effort (or let the opportunity pass) are decisions that must be made carefully. Becoming involved too late or trying to use the wrong method or tool just won't work. Choosing the appropriate model levels and development approach is fairly straightforward for a top-down project like the one portrayed here. It is considerably more difficult for

Figure 9.
IDEF1X Cardinality



a bottom-up effort (for example, installing a software package) or middle-out project (revising or extending an existing system where the physical design will be highly constrained by the current design). Few opportunities are top-down; the majority it seems, are bottom-up or middle-out.

The Tools

Topics like CASE, repositories, object-oriented systems, and so on seem to provoke endless discussion these days. These emerging technologies are promising, but they seem to be surrounded by hype and confusion.

CASE tools promise to automate the transformation of requirements statements into databases and code. I think of them as the means to capture information for each cell in the Zachman framework, reconcile it to information for adjacent cells, and transform it into descriptions at lower-level cells (or, in the case of reverse engineering, to go in the opposite direction). These tools are still in their infancy.

Specific tools fit specific needs, but no completely integrated set is currently available. Some of these tools allow system requirements and solutions to be specified more precisely than others, but the transformation function is, for the most part, still human-assisted.

DB2 rel. 2 has received a lot of attention for its ability to enforce referential-integrity rules (at least in the areas of Insert, Replace, Delete, and primary key/foreign-key enforcement). That's surprising because other DBMSs' relational products already had this ability. Features such as domain and business-rule support are still missing, however.

The object-oriented systems just beginning to emerge are also promising. We'll have to wait until they mature sufficiently for large-scale use.

We must be very careful when choosing tools. Wherever the tool cannot capture or enforce a business rule (cardinality enforcement in relational systems, transformation of functional rules into data value dependencies, and transformation and reconciliation of models and object definitions in CASE tools), code must be written or people relied on to patch the holes. Tools that lack the ability to capture and use necessary information, guarantee its integrity, or integrate with other needed tools should be viewed with considerable skepticism.

The information repository is one area in which this ability is extremely critical. Just what such a repository will look like and how it will be constructed is the cause of much confusion. Divergence of both the standards and, at times, the objectives of the repository's designers is creating chaos (see Paul Winsberg's "Dictionary Standards: ANSI, ISO and IBM," *InfoDB* 3(4), Winter 1988-89).

DA can play a significant role in choosing the right tools and ensuring they will work well together. For example, mapping a tool set onto the Zachman framework can help identify areas that have inadequate tool support or inconsistent support by competing tools.

This might at first seem like a trivial exercise, but you'll be surprised at how difficult it is, especially in large organizations. Why? Because it will

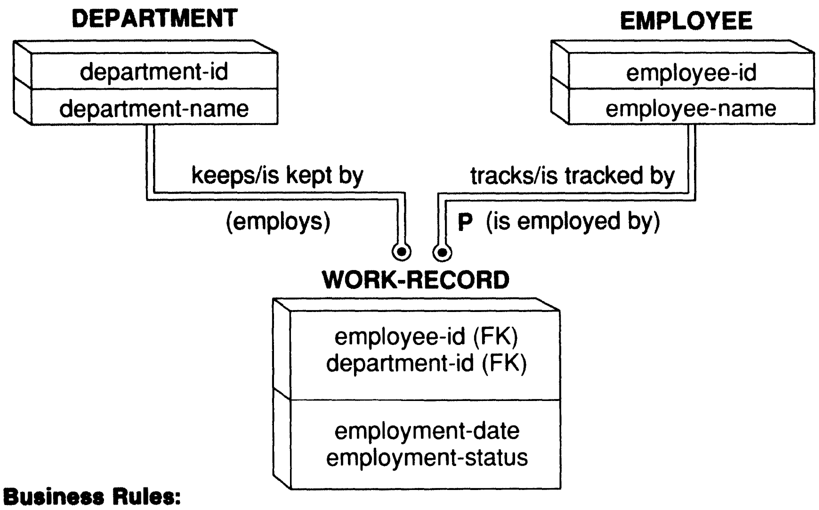


Figure 10.
Many-to-Many Relationships

Business Rules:
A DEPARTMENT keeps zero or more WORK-RECORDs.
Each WORK-RECORD tracks one EMPLOYEE.
An EMPLOYEE is tracked by one or more WORK-RECORDs.
Each WORK-RECORD is kept by one DEPARTMENT.
or
A DEPARTMENT employs zero or more EMPLOYEEs.
An EMPLOYEE is employed by one or more DEPARTMENTs.
A WORK-RECORD records employment of an EMPLOYEE by a DEPARTMENT.

raise political and "turf" issues, not just technical ones. Technical issues are relatively simple; the others are not.

Specification Languages

Our tools are not perfect; they lose some meaning along the way. How about the languages we use to specify our requirements and designs? Some of these are pretty good, especially those that support various levels of the Zachman framework's Function column. A specification language needs to be able to precisely record our understanding of requirements and make information available to tools that will transform this specification into a design. Like the tools, the languages seem to lose some knowledge along the way.

For the data column, the IDEF1X graphic language for information models is one of the best around, but it has its limitations. For each deficiency in a specification language, knowledge is lost.

IDEF1X supports various entities (identification-independent and identification-dependent), relationships (identifying and nonidentifying), and cardinality (required, optional, one to n, and so on) quite well (Figures 8 and 9). Key attributes are shown above the line dividing the entity shape, while nonkey attributes are shown below the line.

Many-to-many relationships can also be shown. Figure 10 brings up the first concern. There are two equally valid ways to record business statements describing this relationship. The shape of the structure is the same but the business rules seem to be different, so you have to choose which verb phrases to use. That's not too difficult if everyone chooses the same style. The integrated computer-aided manufacturing definition (IDEF) modeling tools I'm familiar with cannot discern that the alternatives in Figure 10 are saying the same thing, but they can at least capture either set of relationship statements. Little knowledge is lost, but some ambiguity is introduced. Business-rule summarization functions, such as those available in DACOM's Leverage, also begin to have some trouble (though overall Leverage does an excellent job).

The language and tools can support simple entity subtype structures (referred to as *generalization hierarchies* in this discipline). The right side of Figure 11 shows a simple exclusive OR in which a BUSINESS-PARTY must be either an EMPLOYEE or a DEPARTMENT but not both. More

Figure 11.
IDEF1X Category Structure

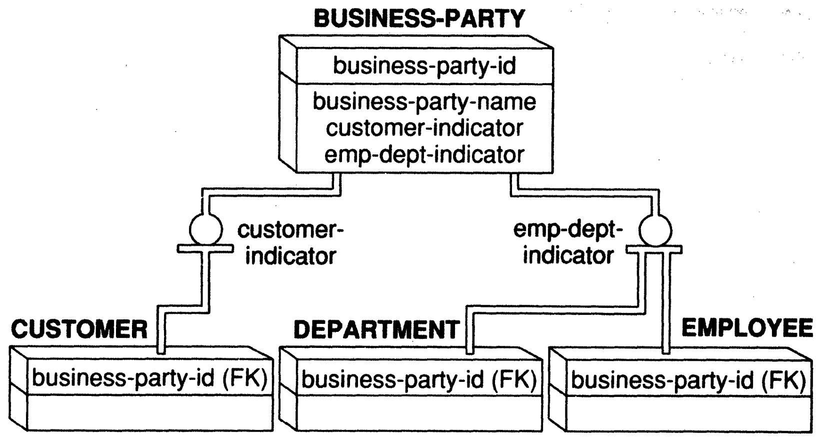
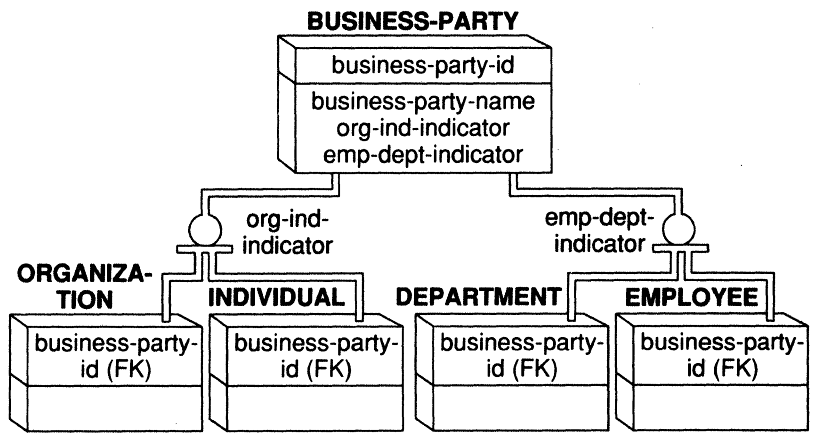


Figure 12.
Complex and/or

The problem is that an individual cannot be a department.



complex category structures are also possible. The left side of the figure shows an additional category, DEPARTMENT, resulting in an AND and OR combined (in addition to being an EMPLOYEE or a DEPARTMENT, a BUSINESS-PARTY can also be a CUSTOMER).

Figure 12 illustrates another problem: It still shows that a BUSINESS-PARTY must be either an EMPLOYEE or a DEPARTMENT but also says that it must be either an INDIVIDUAL or an ORGANIZATION. If we add the constraints that an INDIVIDUAL cannot be a DEPARTMENT but that an ORGANIZATION can be an EMPLOYEE, we run into trouble. I have not been able to find a method to declare this to be true in IDEF. (If you disagree with the business statements implied, turn this into a theoretical problem by substituting A, B, C, and so on for the entity names.) Nor have I been able to find the solution using Chen or Bachman notations. The problem is that the graphic languages cannot, at this point, say everything we need to say.

To fix the languages, we need to extend the graphic structure or, as DA is attempting, add a constraint-specification level separate from the graphics. Constraint specification is a known issue with IDEF1X, as with other languages, and has been around for years.

Situations involving Boolean constraints don't arise often—at least in high-level models—and we can always capture the missing constraints in text notes. But as soon as we have to capture such information in textual form, we relinquish the opportunity to use our automated tools to drive from specification to implementation. The tools cannot read, understand, or use the text.

Perhaps we should add constructs to the graphics to capture the missing information. Or perhaps we should state the constraints as rules

Figure 13.
Alternate Keys

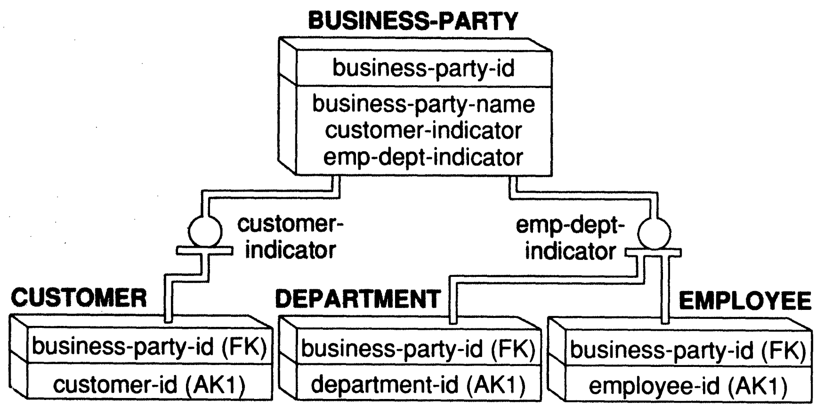
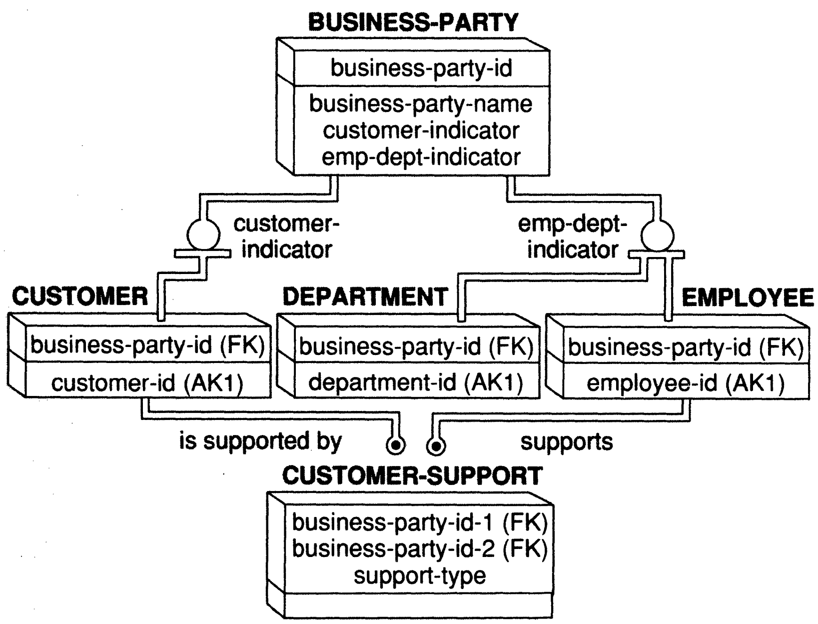


Figure 14.
Alternate-Key Propagation

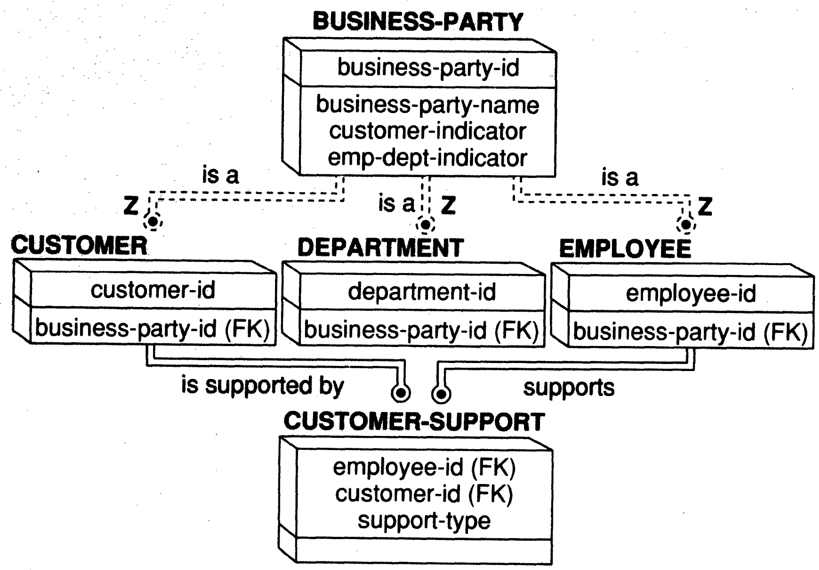


governing the insertion of instances of the various category entities. The choice is not entirely clear. Continuing to add symbols to the graphics will eventually lead to diagrams that try to show so much information that the users' eyes glaze over. Deciding where to stop extending the graphics appears, at least at our current level of maturity, to be a judgment call. However, we must be attentive to situations like this one where decisions made now will determine our success down the road.

In general, IDEF1X handles alternate keys very well. Figure 13 extends the example that uses BUSINESS-PARTY, EMPLOYEE, DEPARTMENT, and CUSTOMER. The last three, being category (subtype) entities, inherit their primary keys from BUSINESS-PARTY. But suppose the business has decided to identify the CUSTOMER manifestation of BUSINESS-PARTY with an attribute called customer-id, the EMPLOYEE manifestation with employee-id, and the DEPARTMENT manifestation with department-id. Also suppose that each attribute is to be designated as an alternate key. So far, so good.

But something funny happens when CUSTOMER and EMPLOYEE become involved in a relationship (no, not conflict of interest): The keys don't propagate correctly (Figure 14). The entity CUSTOMER-SUPPORT will contain a pair of business-party-ids, when we wanted to see a

Figure 15.
Alternate-Key Substitution Work-Around



customer-id and an employee-id. We cannot propagate alternate keys in IDEF1X, so we have to fake it if we want to see the right keys in CUSTOMER-SUPPORT.

Figure 15 shows this "force." Now the keys are right, but again we lose something. The constraint that a BUSINESS-PARTY cannot be both an EMPLOYEE and a DEPARTMENT has been lost as a consequence of the force.

These are situations where the language just can't say it.

The Practitioners

I think it was Yogi Berra who once said that most human endeavors involve people. Once again, he got it right.

Like any business, DA must be careful to choose the right people. Different skill sets are needed for the different functions; you wouldn't necessarily want an expert data-dictionary technician to conduct an information-modeling session or a modeler to tinker with the innards of the dictionary. DA adds value by fitting the right skills to the right tasks and outfitting the right people with the right skills.

Many years ago, while conducting a requirements interview, I tried to get the product managers of a business area to tell me what the system needed to do. Being much younger (I knew more in those days), I became frustrated when I heard things like "We want the capability to modify the rules and change the product very quickly" and "We aren't sure now what the product is supposed to do, but we'll know it when we see it." Come on, guys, make up your minds! What is it you want?

The problem was that they had told me what they wanted, but I was too smart to understand it! I was looking for specific requirements for the product where there weren't any. I didn't recognize that what they needed was a way to look at alternative methods for building potential products. Needless to say, the effort was not a raging success.

We need to be attentive to situations like this. We have to find new ways to think. The ability to think clearly about where the DA discipline is and where it is headed is a key to the data administrator's success. We must add value by keeping a clear vision of where we are trying to take the organization, but we must also maintain a perspective that keeps us from getting too excited about every new trend. We need to keep our focus on short-term value, the mundane stuff like populating a dictionary and digging around in the spaghetti of old systems.

Data administrators add value by helping to obtain specifications, assisting with the transformation of specifications into designs, and patching the holes in the tools and languages. We also add value by seeing across the organization and promoting the reuse of architecture products.

We add value when we understand how much is enough and when we exercise good business judgment. We add value when we focus on the business; we do not add value when we focus on DA or become preoccupied with theoretical perfection.

And we must be ready for change.

Just when we think we might have a grasp on the situation, it blurs on us. For example, John Zachman's framework has been a very useful construct to guide our thinking and discussions. But what happens in the object-oriented world when the Function and Data columns begin to merge into Object? I guess we'll find out.

A Final Word

Our ability to recognize requirements and needs and conceive of new approaches will determine the value DA can add to an organization. If our tools can't use our specifications or our people can't recognize them, we won't succeed. Every now and then we should ask ourselves four questions: How do we perceive ourselves? How are we perceived by our business colleagues? How do we need to manage that perception? How do we operate as a business?

This report has been a somewhat long-winded way of reminding us all of the obvious: that value, like beauty, is in the eye of the beholder. In our business, value must be perceived by the owners, designers, builders, and operators of our information systems.

References

[1]The same basic example is used in training material copyrighted by D. Appleton and Co. Lest I be accused of stealing, it is referenced here. I had been using it for years before I ever saw the company's material and have seen it elsewhere, so I guess that just means it has turned out to be useful.

[2]Zachman, J.A. "A Framework for Information Systems Architecture." *IBM Systems Journal* 26(3), 1987.

[3]This diagram was developed jointly with my friend and colleague Terry Moriarty and is described in her unpublished work entitled "Strategic Information Planning" (July 1989). ■

The Project Management Process

In this report:

Datapro Summary

As its most simple form, project management is a social process in which an objective is completed by performing a sequence of tasks within budget constraints. However, the dynamic nature of people and projects themselves make a project difficult to manage. Organizing, motivating, and directing are all part of a project manager's responsibility.

Project management is a series of actions or activities embodied in a process of getting things done on a project by working with members of the project team and other people in order to reach the project schedule, cost, and technical performance objectives. This description helps to identify project management but it does not tell too much about how a project manager reaches project goals and objectives. This report will describe the project management process along with the idea of the "life cycle" concept.

Project managers get things done by working with people and other resources. They do not perform the operations themselves; rather they manage the members of the project team and other people who do the technical work on the project. Project management is a social process which comprises a series of actions that lead to the accomplishment of project objectives. It is a social process because the actions undertaken by the project managers are principally concerned with relations among people. Management often is viewed as a process.

This Datapro report is a reprint of Chapter 2, "The Project Management Process," pp. 21-36, from *Project Management: Strategic Design and Implementation* by David I. Cleland. Copyright © 1990 by TAB Professional and Reference Books. Reprinted with permission.

Project management is a continuing process. New demands always are put on the project team and have to be coordinated and motivated by the project manager through a process of planning, organizing, motivating, directing, and controlling. As new needs come up before the project, someone has to satisfy these needs, solve the problems, and exploit the opportunities. The project originates as an idea in someone's mind, takes a conceptual form, and eventually has enough substance that key decision makers in the organization select the project as a means for executing elements of strategy in the organization. In practice, the project manager must learn to deal with a wide range of problems and opportunities, each in a different stage of resolution and each having different relationships with the evolving project. This continuing flow of problems and opportunities underscores the need to comprehend a *project management process* which, if effectively and efficiently planned for and executed, results in the creation of project results which complements the organizational strategy.

Managing a large project is so complex that it is difficult to comprehend all of the actions that have to be taken to successfully plan and execute the project. We need to divide the project into parts in order to grasp the full significance of each part and just where that part fits in the scheme of the project. We have to look at the project parts, its "work packages," its logical flow of activities, the phases that the project goes through in its evolution, growth and decline.

A highly useful way of looking at the management disciplines is to examine it in terms of the major functions or processes that are undertaken by the project team under the leadership of a project manager working within the context of a management system.

Management Functions/Processes

A management system may be defined as a regularly interacting or interdependent group of components forming a unified totality. One way of looking at a management system is to view the major activities or functions that are carried out by a manager. Among management scholars and practitioners there is a good deal of difference of opinion as to the interpretation of these major functions. The management functions that are most often associated with the activities of a manager are: planning, organizing, controlling, communicating, actuating, directing, staffing, innovating, representing, creating and motivating.

In this report we will discuss the following major activities of the project manager:

Planning: is the definition of the purposes of the project and a prescription of how that organization is going to use resources to attain its purposes.

Organizing: is the determination of the quality and quantity of resources that are needed on the project and the procurement, alignment and development of these resources.

Motivating: is the creation of an environment which provides people on the project with the opportunity to gain maximum psychological, social, and economic satisfaction in their place of employment.

Directing: is the face-to-face supervision of other people, including the ability to provide leadership for people working on the project to accomplish their personal purposes as well as the project purposes.

Controlling: includes the development of performance standards, the sensing of individual and organizational performance, the comparison of actual with planned performance, and the institution of corrective action to reprogram, replan, and realign the use of resources when the project fails to meet the project performance standard. Taken together, these functions can be treated as a management system composed of planning systems, organizational systems, motivational systems, leadership systems, and control systems integrated in a project management process to bring about a successfully completed project.

In Table 1, the project management process is portrayed in terms of its major functions. The activities noted under each of these functions are only representative. Effective project management requires many more activities under each of these functions.

An important part of any project management process is recognition of the management processes that are involved in each of the phases a project goes through during its life cycle.

Table 1. Representative Functions/Processes of Project Management

Planning (What are we aiming for and why?)

- Develop project objectives, goals, and strategies.
- Develop project work breakdown structure.
- Develop precedence diagrams to establish logical relationship of project activities and milestones.
- Develop time-based schedule for the project based on the time precedence diagram.
- Plan for the resource support of the project.

Organizing (What's involved and why?)

- Establish organizational structure for the team.
- Identify and assign project roles to members of the project team.
- Define project management policies, procedures, and techniques.
- Prepare project management charter and other delegation instruments.
- Establish standards for the authority, responsibility and accountability of the project team.

Motivation (What motivates people to do their best work?)

- Determine project team member needs.
- Assess factors that motivate people to do their best work.
- Provide appropriate counseling and mentoring as required.
- Establish rewards program for project team members.
- Conduct initial study of impact of motivation on productivity.

Directing (Who decides what and when?)

- Establish "limits" of authority for decision making for the allocation of project resources.
- Develop leadership style.
- Enhance interpersonal skills.
- Prepare plan for increasing participative management techniques in managing the project team.
- Develop consensus decision making techniques for the project team.

Control (Who judges results and by what standards?)

- Establish cost, schedule, and technical performance standards for the project.
- Prepare plans for the means to evaluate project progress.
- Establish a project management information system for the project.
- Prepare project review strategy.
- Evaluate project progress.

Project Life Cycles

This section of the report examines some of the characteristics of a project that are dynamic. This dynamism is manifest in a project, its life cycle, the organizational underpinnings that support the project, the fluid assignment of members on the project team, and the continuing interface of organizational elements.

Projects, like organizations, always are in motion as each proceeds along its life cycle. Projects go through a life cycle to completion, hopefully on time, within budget, and satisfying the technical performance objective. When completed, the project joins an inventory of capability provided by the organization that owns the project.

All projects—be they weapons systems, transportation systems, or new products—begin as a gleam in the eye of someone and undergo many different phases of development before being deployed, made operational, or marketed. For instance, the U.S. Department of Defense (DOD) uses a life-cycle concept in the management of the development of weapons systems and other defense systems. A U.S. Air Force version of this life cycle identifies a number of phases, each with specific content and management approaches. Between the various phases are *decision points*, at which an explicit decision is made concerning whether the next phase should be undertaken, its timing, etc. Generically, these phases are:

The Conceptual Phase: During this phase, the technical, military, and economic bases are established, and the management approach is formulated.

The Validation Phase: During this phase, major program characteristics are validated and refined, and program risks and costs are assessed, resolved, or minimized. An affirmative decision concerning further work is sought when the success and cost realism become sufficient to warrant progression to the next phase.

The Full-Scale Development Phase: In the third phase, the design, fabrication, and testing are completed. Costs are assessed to ensure that the program is ready for the production phase.

The Production Phase: In this period, the system is produced and delivered as an effective, economical, and supportable weapons system. When this phase begins, it denotes that the weapons system has reached its operational ready state and is turned over to the using command. During this period, responsibility for program management is transferred as an Air Force logistics supporting capability within the Air Force.

The Deployment Phase: In this phase, the weapons system is actually deployed as an integral organizational combat unit somewhere within the Air Force.

The project development life-cycle concept recognizes a natural order of thought and action pervasive in the development of many kinds of projects, whether commercial products, space exploration projects, or management projects.

New products, services, production processes or roles for the organization have their genesis in ideas evolving within the organization. Typically, such ideas go through a distinct life cycle, i.e., a natural and pervasive order of thought and action. In each phase of this cycle, different levels and varieties of specific thought and action are required within the organization to assess potential efficacy of the project.

The material that follows lists some of the key considerations that must be evaluated in each of the several phases of the life cycle. These considerations, when fully evaluated, provide the basis for the investigation of the many forces and factors that can have an impact on the project. Each of these considerations should lead to a full investigation of subissues and problems or opportunities that can have an impact on the project, its outcome, and the future strategies of the organization.

The phases of a project are somewhat dependent on the type of project. For example, an engineering project's phases would include: research and development, design, construction or manufacturing, implementation and maintenance, and transfer of technology. Figure 1 compares the terminology of the generic project life cycle and the phases of an engineering project.

The germ of the idea for a project may evolve from other research, from current organizational problems, or from the observation of organizational opportunities. The conceptual phase is one in which the idea is conceived and given initial assessment.

During the conceptual phase, the environment is examined, forecasts are prepared, objectives and alternatives are evaluated, and the first examination of the technical performance, cost, and time objectives of the project is performed. Preliminary strategy, organization, and resource requirements are conceived. A feasibility study is conducted of the project requirements in order to provide a basis for further detailed evaluation. Table 2 shows representative major details of these efforts.

The Conceptual Phase

During the conceptual phase, the mortality rate of projects is high. The study process conducted during this phase should identify projects that have high risk and are technically, environmentally, or economically infeasible or impractical.

The conceptual phase provides the opportunity to determine if the project truly promises to be worth doing in support of organizational strategies. The better the analysis conducted during the conceptual phase, the more opportunities for uncovering intelligence on how the project can best be managed in its future life cycle phases. Senior managers in the enterprise owning the project have a special responsibility to evaluate the promise a project holds to bring about the needed change in the enterprise's future. The board of directors, which acts in the capacity of trusteeship in prudently and economically managing corporate assets for the benefit of stockholders, has the ultimate authority and responsibility for study and approval of key proposals involving corporate strategy, particularly those risky projects that involve a substantial portion of corporate resources. The board of directors also has the respon-

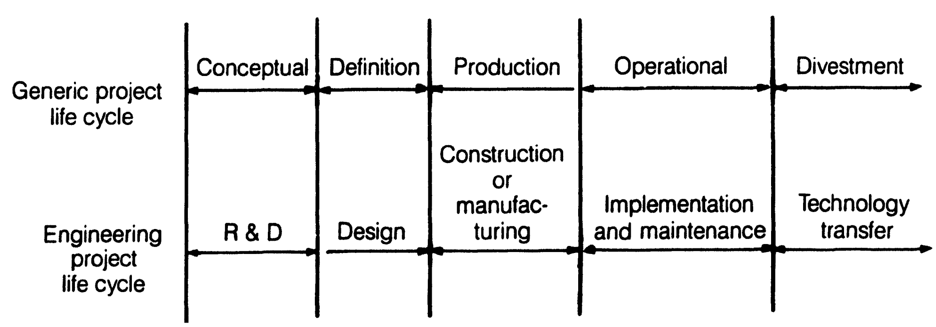


Figure 1.
The Flow of the Design Development Process

Table 2. Conceptual Phase of Project

1. Determine existing needs or potential deficiencies of existing organizational capabilities.
2. Establish strategic concepts which provide initial guidance to overcome existing or potential deficiencies.
3. Determine initial technical, political, legal, social, competitive and economic feasibility and practicability of the project.
4. Examine alternative ways of accomplishing the project objectives to include alternative projects.
5. Provide initial answers to the questions:
   - What will the project cost?
   - When will the project be available?
   - What will the project do?
   - How will the project be integrated into existing systems?
6. Identify the human and nonhuman resources required to support the project.
7. Select initial project designs which will satisfy the project objectives.
8. Determine initial project interfaces.
9. Establish a project team organization.
10. Determine the "strategic fit" of the project, i.e., How will the operational use of the project be determined; how well does the project play its role as a means for executing the strategy of the project owner.

sibility to maintain strategic surveillance over a capital project during its life cycle.

Once the conceptual phase of the project is completed and the preliminary information is developed, or, when not complete, a full understanding of what additional information is required is understood by the project team, then the project can take a step forward in its life cycle.

Definition Phase

The definition phase simply tells in more detail what it is we want to do, when we want to do it, how we will accomplish it, and what it will cost. The purpose of the definition phase is to determine the projects cost, schedule, performance objectives, and resource requirements, and whether all work packages, projects and organizational strategies fit together economically and technically. The definition phase allows the organization to fully conceive and define the project before it commits too many resources to the project. Simply stated, the definition phase dictates that we stop and take time to see whether this is what we really want before the resources are committed to putting the project into production and operation. If the idea has survived the end of the conceptual phase, the organization gives a conditional approval for further study and development. The definition phase provides the opportunity to review and confirm the decision to continue design and development, perhaps create a prototype, and make a production or construction decision.

Senior managers' opportunity to influence the project and its strategic outcome in the organization are greatest during the conceptual and definition phases. Once the project progresses beyond these phases organizational resources have been committed, supporting strategies have been synchronized, and terminating of the project is more difficult. Surveillance of the project during these phases is critical to determine if the probable outcomes of the project fit into the probable future of the organization. Cost, schedule and technical performance risks and uncertainties are greatest during these phases. Decisions made during and at the end of the definition phase might very well cancel further work on the project and redirect organizational resources elsewhere. The elements of this phase are described in Table 3.

Key outputs of the project definition phase are: First, the final agreement on the design cost, schedule, and technical performance objectives of the project; second, the suitability of the fit of the project in the enterprise strategy; third, the availability of resources, actual or planned, to support the project during its life cycle; and finally, the adequacy and suitability of a project management system to be used as a philosophy and guide to the management of the project during its life cycle.

Accuracy in the project design is a key consideration of the life cycle of the project. When cost overruns happen, an important alternative to consider is to re-engineer the project to identify and correct original design errors.

Assuming everything is ready and the criteria of the definition phase have been satisfied, then the project can progress to the next phase—production and/or construction.

Production Phase

The purpose of the production or construction phase is to acquire and test the project elements and the support systems using the standards developed during the preceding phases. The production phase involves such things as the installation of the project, the fabrication of hardware and software for the construction of facilities, and the finalization of supporting technical and training documentation.

Quality assurance practices are important in this phase. Manufacturing efficiency and the productive use of all resources can make the difference between profit and loss on the project's product. Table 4 shows the important elements in the production or construction phase.

Operational Phase

As the product or service resulting from the project is produced or installed the operational phase is initiated. The customer for whom the project is intended receives the

Table 3. Definition Phase of Project

1. Firm identification of the human and nonhuman resource required.
2. Preparation of final project performance requirements and the project design.
3. Preparation of detailed project plans required to support the project.
4. Determination of realistic cost, schedule, and performance requirements.
5. Identification of those areas of the project where high risk and uncertainty exist, and delineation of plans for further exploration of these areas.
6. Definition of intersystem and intrasystem interfaces.
7. Determination of necessary support subsystems.
8. Identification and initial preparation of the documentation required to support the project, such as project manager charter, policies, procedures, job descriptions, budget and funding papers, letters, memoranda, etc.
9. Final design and development of the Project Management System to support the project to include (1) the facilitative matrix organization; (2) the cultural ambience; (3) the human subsystem; (4) the planning subsystem; (5) the project management information subsystem; (6) the project control; and (7) the selection of contemporary state-of-the-art project management concepts, processes, techniques and methodologies.

Table 4. Production/Construction Phase of Project

1. Updating of detailed plans conceived and defined during the preceding phases.
2. Identification and management of the resources required to facilitate the production or construction processes, such as inventory, supplies, labor, funds, etc.
3. Verification of production specifications.
4. Beginning of production, construction, and installation.
5. Preparation and dissemination of suitable policy and procedural documents.
6. Performance of final testing to determine adequacy of the project or service to do the things it is intended to do.
7. Development of technical manuals and affiliated documentation describing how the project product or service is intended to operate.
8. Development of plans to support the product or service during its operational phase.

product and puts it to use. In some cases the organization that produced the project will provide maintenance and service support through a contract with the project user. Elevators, machine tools, computers, and building energy management systems are some examples where such maintenance and service contracts are used.

During the operational phase, the project usually loses its identity per se, and is assimilated into the ongoing business of the user. If the project leads to a product to be marketed, the operational stage begins the sales life-cycle portion of the overall life cycle, for it is in this phase that marketing of the product is conducted. Table 5 lists some of the key elements of operational phase.

The products and/or services created by a product usually become obsolete or wear out, or other changes motivate a customer to seek newer products or services. However, buildings, bridges, highways or a technology developed through a project management approach can endure for centuries, such as the Great Pyramids, or the ancient buildings constructed in Rome and Athens.

Divestment Phase

The divestment phase is the one in which the project team is out of the business which it began with the conceptual phase. Every system—be it a product system, a weapons system, a management system, or whatever—has a finite lifetime. Too often this goes unrecognized, with the result that outdated and unprofitable products are retained, inefficient management systems are used, or antiquated equipment and facilities are "put up with." Only by the specific and continuous consideration of the divestment possibilities can the organization realistically hope to avoid these contingencies. Table 6 relates to the divestment phase.

Taken together, Tables 2 through 6 provide a detailed outline of the overall project development life cycle. Of course, the terminology used in these tables is not applicable to every project, because different industries use different terminology. However, whatever the terminology, the concepts are applicable to all such projects.

Table 5. Operational Phase of Project

1. Use of the project results by the intended user or customer.
2. Actual integration of the project's product or service into the customer's organizational systems.
3. Evaluation of the technical, social, and economic sufficiency of the project to meet actual operation conditions.
4. Provision of feedback to organizational planners concerned with developing new projects.
5. Evaluation of the adequacy of supporting systems such as maintenance, service and other logistic support.
6. Training of user personnel.

Managing the Life Cycle

One of the first undertakings in planning for a project is to develop a rough first estimate of the major tasks or work packages to be done in each phase. Once established, the life cycle estimate should be updated as more is learned about the project. As the project progresses through its life cycle, it exhibits ever-changing levels of cost, time, and performance. The project manager must make correspondingly dynamic responses by changing the mix of resources assigned to the project as a whole and to its various work packages. Thus, budgets will fluctuate substantially in total and in terms of the allocation to the various project work packages. The need for personnel and various kinds of expertise will similarly fluctuate, as will virtually everything else.

This constantly changing picture of "peaks and valleys" is an underlying structural rationale for project management. The traditional hierarchical organization is not fully designed to cope with managing such an always-changing mix of resources. Rather, it is designed to control and monitor a much more static entity that, day-to-day, involves stable levels of expenditures, numbers of persons, etc.

Project Life Cycles and Uncertainty

Many organizations can be characterized at any instant by a "stream of projects" that place demands on its resources. The combined effect of all the projects facing an organization at any given time determines the overall status of the organization at that time.

The projects facing a given organization at a given time typically are diverse—some products are in various stages of their life cycles, other products are in various stages of

Table 6. Divestment Phase of Project

1. Project phasedown.
2. Development of plans transferring responsibility to supporting organizations and user.
3. Divestment of transfer of resources to other projects.
4. Development of "lessons from project" for inclusion in qualitative-quantitative data base for management of future projects.
   - Assessment of project management image by the user.
   - Major problems encountered and their solution.
   - Technological advances.
   - Advancement in knowledge relative to department strategic and project objectives.
   - New or improved management techniques.
   - Recommendations for future research and development and project management applications.
   - Recommendations for the management of future projects.
   - Other major lessons learned during the course of the project life cycle.

development. Management subsystems are undergoing development. Organizational subsystems are in transition. And major decision problems, such as merger and plant location decisions, sometimes are studied as projects.

Moreover, at any given time, each of these projects usually will be in a different phase of its life cycle. For instance, one product may be in the conceptual phase undergoing feasibility study, another may be in the definition phase. Some might be in production. Others possibly are being phased out in favor of oncoming models.

The challenges associated with the overall management of an organization that is involved in a stream of projects are influenced by life cycles just as are the challenges associated with managing individual projects. In project-driven organizations whose main business is management of the stream of projects passing through the organization, the mix of projects in their various phases is most challenging, particularly in allocating personnel, funding resources, scheduling workloads, etc., to maintain a stable organizational effort.

Summary

This report dealt with the dynamic nature of projects and project management. As such, it is the linking mechanism between the environment, which creates the need for project management, and the specific organizational and management techniques of project management that were briefly introduced here.

Project management is a social process of working with people through the application of the key management functions of planning, organizing, motivating, directing, and controlling.

Project management is an approach for responding to the dynamic nature of the flow of projects in an organization. Since complex projects are part of strategies to deal with complex problems of organizations, there is a real need to develop management techniques and devices which address themselves to the dynamic nature of projects. ■

Information System Cost Estimating

In this report:

This report will help you to:

- Identify several potential causes of problems in cost estimating and take measures to overcome them.
- Develop more accurate cost estimates for information system projects.

Abstract
Information systems cost estimating is an important management concern. An estimate helps to cost justify individual proposals, to schedule their development, to staff them, to control and monitor their progress, and to evaluate estimators and implementers. Through a case study of a chemical manufacturer, the investigation described in this report facilitates a better understanding of the management of the cost estimating process. Interviews with 17 information systems managers and staff members, and four user managers confirm that the practice of cost estimating can be viewed in terms of both a Rational Model and a Political Model, can identify impediments to accurate estimating, and can provide suggestions and warnings for managers and future researchers.

Introduction
In principle, the expected economic value of a proposed information system governs the decision to develop it. An organization conventionally predicts the benefits of the system over its lifetime and estimates the costs of its initial development and ongoing operations to identify that value.[1,2] Such financial techniques as net present value, return on investment, or internal rate of return are often used to combine and summarize tangible benefits and costs.[3] Ideally, a steering committee of senior managers evaluates these calculated figures, along with any available intangible values, and plans a portfolio of applications.[4,5]

The accurate prediction of benefits and costs is crucial because of this approval process. Overestimated

This Datapro report is a reprint of "Information Systems Cost Estimating: A Management Perspective" by Albert L. Lederer, Rajesh Mirani, Boon Siong Neo, Carol Pollard, Jayesh Prasad, and K. Ramamurthy, pp. 159-176, from *MIS Quarterly*, Vol. 14, No. 2, June 1990.

benefits or underestimated costs can result in the decision to develop projects that will ultimately fail to contribute to the organization and will waste scarce resources. Underestimated benefits or overestimated costs can result in the decision to refrain from developing potentially worthwhile projects. Thus, the failure to accurately assess all of the benefits and all of the costs can have a major, widespread impact on the organization.

The impact of inaccurate estimating has been so significant that it has reached the popular business press.[6] This *BusinessWeek* article reports that a state of Oklahoma project was initially estimated at $500,000 but was later completed at $4 million. It also describes an Allstate insurance system that was estimated at $100 million while under development; it was initially estimated at $8 million. In fact, a recent Peat Marwick Mitchell & Co. survey found that 35 percent of its largest clients admitted major cost overruns.[6] Such catastrophes have destroyed the credibility of ISD management.

The most significant cost is the expenditure for the labor (i.e., systems analyst and programmer time) used to develop an information system.[7] One study identifies the estimation of labor requirements as an important ISD planning issue among top ISD managers.[8] That research found that these managers are concerned about the estimation of labor requirements because they must use these requirements to compute overall ISD development costs. That study instigated the research described in this report whose objective was to better understand the management of the labor cost estimation process through the in-depth study of the techniques and problems of one organization.

Background

Previous research regarding cost estimation has largely focussed on the study of algorithmic methods. These methods generally have three major features.

First, they require some well-defined level of knowledge of the proposed system. That is, the methods demand the identification of various, relevant parameters. For example, many methods require that the estimator identify the number of lines of executable code in the proposed system.[9-16] Other techniques focus on the functions of the proposed system and might require such parameters as the number and complexity of inputs, inquiries, outputs, and master files.[17-20] One particular method asks the estimators to rate such diverse characteristics (among many others) as the developers' familiarity with the application, the quality of the analysis of the project objectives, and the users' knowledge of data processing.[21]

Second, the methods generally require some historical base of information about past projects. This information includes durations of projects as well as some of the parameters mentioned above.

Third, the methods provide formulas that forecast the cost of the proposed system. The formulas are based on the system's parameters and on the corresponding parameters and costs of the systems in the historical base of information. The formulas range from simple to complex in mathematical sophistication. For example, some utilize such simple summary statistics as means and standard deviations,[18] while others employ differential equations.[14]

Objective studies of these methods have been few. Often the developers of a method describe their own technique and report their own assessment of its accuracy.[12, 14-16, 18] Other research has used the investigator's full knowledge of the scope of completed projects to develop a formula that would have predicted the projects' durations had the projects not been completed and their durations not already known. (In contrast, one might expect the investigator to test a proposed formula by predicting the duration of recently conceived projects and, after their completion, contrasting the predicted and actual durations.)

For example, one study evaluated the accuracy of four algorithmic methods by predicting the durations of projects that had already been completed.[22] However, it found considerable inaccuracy with error rates ranging from 85 percent to 772 percent.

Nevertheless, the logic of the algorithmic methods is impressive. In fact, it has inspired the view that "the methods available today are more than adequate to establish an estimation approach. All that is needed is management's willingness to employ the planning and control philosophy used in other functional areas in the information systems department."[23] In attempting to understand the management of the estimating process, this study recognizes that there is a question of the adequacy of algorithmic methods versus the adequacy of the application of planning and control. It offers a new perspective on that question by focusing on a management point of view.

Conduct of the Study

This study employs a case methodology.[24] A case methodology is appropriate in research such as the management of cost estimating, where theory and research are in their early, formative stages.[25] Moreover, the management of cost estimating is among those "sticky, practice-based problems where the experiences of the actors are important and the context of action critical"[26] and where a case methodology is warranted.

Accordingly, the authors gained the interest and cooperation of a chemicals manufacturer. The firm is the largest division of a *Fortune* 200 corporation. The division has approximately $1 .6 billion in annual revenue and 6,000 employees. of which over 250 are in the Information Systems Department (ISD). The ISD's annual budget is about $22 million. User departments have a backlog of requests for limited ISD service and must make judicious decisions about proposed systems because their departmental budgets are charged for them. In light of its budget, its size, its well-defined standards and documentation, the experience and education levels of its management and staff, and its extensive computer-based applications, an independent observer would probably consider the ISD to be technically sophisticated and well managed. This is important to remember because the ensuing discussion might make it appear otherwise.

The authors interviewed 17 ISD managers and professionals. During a structured interview, subjects were asked to: (1) identify their job titles, as shown in Table 1; (2) describe their job responsibilities, especially with regard to their roles in the cost estimating process, as shown in Table 1; (3) explain how systems' costs are estimated; (4) identify any potential causes of estimating problems; and (5) identify potential means of improving the estimation process. The authors also interviewed four representatives from the functional areas of the organization with essentially the same questions; these representatives had all participated extensively on ISD development projects. The interviewers supplemented the five main questions with requests for clarification and related information. Each interview lasted about one and a half hours.

Three authors participated in each interview for purposes of control. One particular author was present for all of the sessions, while the remaining authors rotated as the other two interviewers. Immediately after each session, notes were transcribed independently by each interviewer, then compared and cross-checked to develop final versions. Later, the authors independently encoded and summarized all of the subjects answers to the questions.

In addition, the authors obtained and reviewed management reports and data about past projects. A key ISD manager helped them obtain this information, choose interviewees, and verify current findings.

The most comprehensive report showed primarily small projects from January 1 through May 17, 1987. It covered 1,622 hours of labor over the 19-week period. Of all the projects on the report, 71.5 percent were completed beyond ±10 percent of the estimate that appeared on the report. Likewise, of all the projects on the report, 60 percent were completed beyond the wider range of ±20 percent. The greatest overruns included 60 percent, 64 percent, 119 percent, 247 percent, 520 percent, and 533 percent. However, most of those were small projects and hence, overruns or underruns might not be considered as deleterious as on larger projects. Nevertheless, they do suggest a potential problem in the area of cost estimating. In Figure 1, those data points outside the diagonal lines represent the 71.5 percent of the organization's small projects completed beyond ±10 percent of their estimate.

At the conclusion of the study, the authors gave an oral presentation and a written report to a team of ISD managers. The subsequent discussion confirmed that the authors had clearly understood the cost estimating process of the organization.

Two Perspectives

The interviews revealed that ISD management had a documented method for managing the estimation process. Much of it was outlined in ISD's Standards and Guidelines Manual. However, simultaneously, activities not described in the manual also took place. Such parallel, coexisting activities are suggested by Robey and Markus.[27]

Robey and Markus allude to the existence of a rational system development process in which

Table 1. Titles of Respondents and Relevant Responsibilities

ISD Interviewees	
ISD Director	Responsible for entire information Services Department
Manager of Systems Development	Responsible for development of large and some small systems
Manager, Financial Information Systems	Responsible for development of financial systems
Manager, Marketing Information Systems	Responsible for development of marketing systems
Manager, Manufacturing and Engineering Information Systems	Responsible for development of manufacturing and engineering systems
Manager of Systems Service Center	Responsible for small systems development and maintenance projects
Senior Systems Specialist	Serves as project manager for large projects
Systems Specialist	Serves as project manager for large projects
Contract and Financial Administrator	Responsible for financial information for ISD director
Manager of Systems and Programming Support	Provides technical assistance to systems specialists, programmers, etc.
Manager of Planning and Administration	Responsible for long-range information systems planning
Manager of Systems Planning	Responsible for initial planning of large projects
Programmer/Analyst	Responsible for systems development and maintenance
Programmer	Responsible for systems development and maintenance
Contract Programmer	Responsible for systems development and maintenance
Users	
Controller	Chief financial officer and major user
Manager of Cost and Lease Accounting	Liaison for system development for accounting applications
Manager of Purchasing	Liaison for systems development for purchasing applications
Manager of Engineering Systems	Liaison for systems development for engineering applications

ISD and user participants follow a prescribed set of phases to create an information system. They state that a rational process has an identifiable and agreed upon set of goals with a prescribed means to achieve them.

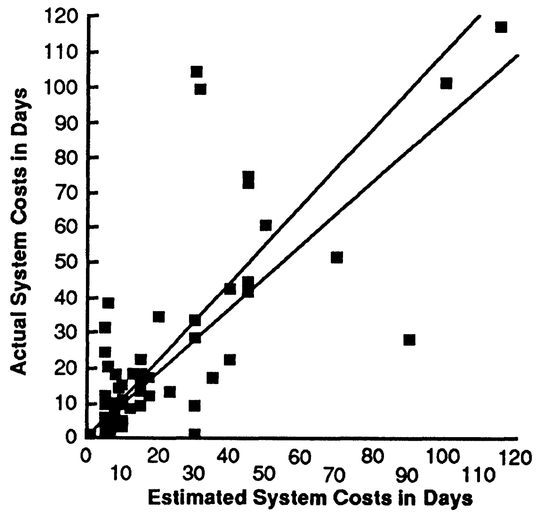
Illustrating the rational process within the area of cost estimating is Jones'[28] technique, called SPQR (software productivity, quality, and reliability). Based primarily on computations using 20 historical factors, SPQR predicts various types of systems development cost estimates and schedules, the rational process's identified and agreed upon goal. Examples of SPQR's major factors are programming languages used, program size, and experience of programmers and design personnel. The use of these factors represents the rational process's prescribed means of achieving the goal.

However, Robey and Markus as well as other earlier researchers[29-31] recognize the inadequacy of a rational process as a means of understanding behavior. For example, Wildavsky[31] analyzed the federal government's budgetary process and found the rational paradigm to be inefficient, ineffective, and inappropriate because most outcomes emerged not so much through the extensive planning that was performed, but rather through considerable bargaining and mutual adjustment.

The inadequacy of the rational model is echoed by Pfeffer[32] who observed that it does not recognize the origin of the personal objectives that lead to conflicts of interest. Moreover, he recognizes that the rational approach does not account for the fact that the outcome of such conflicts can be interpreted in terms of who benefits and who loses.

Hence, Robey and Markus also suggest the existence of a political systems development process in which participants' personal motives inspire their actions. They state that the participants in a political process have differing objectives and may use the situation to achieve their objectives to

Figure 1.
Estimates and Actuals



the disadvantage of each other. Such motive-driven behavior has been helpful in behavioral analysis in both the ISD and other contexts.[29, 33-36]

The political approach recognizes that conflicts of interest are common and expected. It further recognizes the necessity to understand the participants, their stands on the issues, the determinants of their relative power, and their process of arriving at a final decision. Activities result from bargaining and compromise, and the resulting decisions seldom reflect the preferences of a single group.[33] Ultimately, the relative power of the participants provides both the sufficient and necessary means of reaching the decision.[32,37] In fact, with regard specifically to cost estimating, Jones proposes that "perhaps 10 to 15 percent of the really catastrophic project failures are at least partly caused by personal human dislikes and political or territorial disputes"[28] although he does not elucidate such conflicts or formally include them in SPQR.

The next two sections describe the estimation process of the chemicals manufacturer. The first section describes a Rational Model in which the main identifiable and agreed upon goal is to provide an accurate estimate, and the prescribed means is a method of arriving at it. The second section describes a Political Model in which the participants have different objectives and use the cost estimating process to realize them. Each model reveals some of the management problems in estimating information systems development labor requirements.

Evidence of the Rational Model

In the Rational Model, five groups of employees from the chemical manufacturing company participate in systems development activities that are relevant to cost estimating:

1. **ISD estimators:** planners who work within the ISD to prepare initial project proposals, which include projected systems costs.
2. **ISD implementers:** systems analysts and programmers who develop the proposed system. They might also provide revised estimates, but frequently are not the same individuals who initially estimate costs.
3. **ISD management:** executives responsible for the ISD.
4. **User contacts:** liaisons from the application area of the business to the ISD. They might not be actual system users.
5. **User management:** executives responsible for the application area within the business organization.

Under the Rational Model, these managers and their staff carry out the following planning and development phases of a typical systems development life cycle approach.[38]

Phase 1—The Initial Investigation

User management or the user contact generates an idea in this phase. That person or group informally requests a provisional estimate by asking any available ISD representative (ISD manager, estimator, or implementer) for this number. The ISD representative gives a very rough estimate for the sole purpose of enabling the requester to decide whether to proceed with Phase 2. According to a senior systems specialist in the Manufacturing and Engineering Information Systems group, the very rough Phase 1 estimate is often very inaccurate because of the lack of knowledge about the scope of the project by all parties. The manager of Financial Information Systems (and four other interviewees) further pointed out that the ISD representative assumes the estimate will be forgotten when better estimates (i.e., based on more knowledge) are developed later on.

Phase 2—Preliminary System Study
According to the manager of Purchasing (a user), if the user contact believes that the system's benefits can exceed the Phase 1 cost estimate, a preliminary system study is carried out. During this phase, the ISD estimator examines the proposed application in order to produce an initial, serious estimate.

The estimator roughs out the system design to produce this estimate, a process described by the manager of Systems Planning. The estimator first identifies modules of programs with their functions, then determines files, and next assesses the complexity level of each module. Then the estimator looks up the number of days of analysis and programming for each module in an existing matrix of complexity levels and durations. Finally, the estimator sums the number of days and adjusts them based on experience and intuition. This algorithmic technique is hereafter referred to as "rough/refine" by the authors because the initial rough-cut estimate is later refined based on an increased understanding of the application and the user requirements.

The manager of Purchasing (a user) stated in the interview that "there is not enough brain power to catch" all requirements and that hence ISD management, user management, the user contact, and the estimator all recognize that this estimate is based on limited knowledge about the application and the requirements. They further recognize that the estimate will be refined later when there is more information about the application and requirements. Nevertheless, the Phase 2 estimate must be officially approved based on the Standards and Guidelines Manual. ISD management, user management, the user contact. and the estimator all sign off on the Phase 2 estimate to demonstrate their approval.

Phase 3—System Planning
Phase 3 includes a thorough analysis of the system with extensive user participation, and then, based on this analysis, a possible revision of the estimate. In the interview, the manager of Systems and Programming Support attributed this revision to the user's decision to change the requirements, while both the controller and the manager of Purchasing (both users) attributed it to a lack of understanding of the project. The manager of Systems Planning stated that at the beginning of this phase, the estimator gives his or her documentation to the implementer and withdraws from the project. The implementer then learns much more about the proposed system during this analysis.

The ISD implementer applies the rough/refine technique to create the Phase 3 estimate. If the ISD implementer's estimate exceeds the Phase 2 estimate, presumably because of more knowledge about the application and the requirements, then ISD management, user management, and the user contact again must sign off on a form shown in the Standards and Guidelines Manual. If the Phase 3 estimate is exorbitantly high, user and ISD management have the option to cancel the project.

Phase 4—System Design
Phase 4 represents the preparation of detailed system design. In terms of estimating activities, the implementer uses rough/refine once again, but with considerably more information. As in Phase 3, if the Phase 4 estimate changes, signoff takes place. For example. in the interview, the controller described a stock transfer system that had originally been estimated at 100 worker-days but was reestimated at 200 worker-days during the Phase 4 estimate. He threatened to cancel the project but signed off on this estimate instead because ISD told him that cancellation was impossible. Project cancellation does, however, often remain an option.

Phase 5—Program Specifications
The estimate probably remains unchanged while the ISD implementer writes the specifications for programming. If it changes, signs would again be required.

Phases 6-10
The next five phases are: Phase 6—Programming; Phase 7—Implementation Planning; Phase 8—Testing; Phase 9—User Training; Phase 10—Implementation. During these phases, formal estimates are not prepared. However, should an overrun appear imminent, the ISD implementer informs user management and user contacts of the new completion target date.

Evidence of the Political Model
Under the Political Model, the same groups of employees participate in the systems development

process. The Political Model, however, recognizes their motives, objectives, conflicts of interest, and relative power. The groups seek to utilize their power to determine the outcome of the decision process of agreeing on a cost estimate and thus authorizing further system development. Their overriding and governing objective is to make contributions to the organization in order to increase their visibility, demonstrate their competence, and enhance their reputations. By showing that they are performing well, they can increase their responsibilities, their staffs (if they have any), and ultimately their financial compensation (a political strategy akin to the "Reputation Game," as proposed by Bardach[39] and elucidated by Grover, et al.,[35] in which parties attempt to persuade others that they are doing more or better than they actually are and thus increase their power).

Hence, they seek that the outcomes of the decision process facilitate this objective. The specific means of influencing the decision and making their contributions visible may differ depending on their power and their responsibilities. To illustrate this Political Model, in coexistence with the Rational Model, brief examples (of which others might be identified) are given for each group of employees: (The subsequent extended discussion attributes assertions to specific interviewees.)

1. User contacts have the power to understate their needs in order to influence cost estimates. The source of this power is the user contacts' intimate understanding of proposed applications. User contacts might seek the approval of an expensive, new system that would increase the recognition of their contribution to their department. To obtain the approval of the initial proposal, user contacts can understate their needs and thus encourage an estimate low enough to gain approval. (Conversely, with different objectives user contacts can overstate their needs.)
2. User management has the power to sway user contacts to understate (or overstate) their needs in order to influence cost estimates. The source of this power is user management's influence over user contacts' salaries and promotions. If user management seeks to readily obtain more new systems to permit their subordinates to promptly complete more business-related work, they can encourage their user contacts to understate their needs, obtain low estimates, and gain their approval.
3. ISD management has the power to sway estimators to inflate (or reduce) estimates because of their direct influence over estimators' salaries and promotions. ISD management might seek to install a particular new system on time and within budget in order to increase corporate profits because such a contribution to the organization improves their reputation. In that case, ISD management might encourage estimators to inflate an estimate to ensure its timely completion.
4. ISD estimators have the power to use their estimates to convince user management and ISD management to authorize new systems. If authorized and installed, the estimators' contribution will be acknowledged. If proposed systems are not priced to sell (i.e., they are overpriced), managers will not authorize them, they will not be installed, the organization will not achieve their benefits, and little recognition will accrue to the estimators. Estimators who continually evaluate systems, none of which are implemented, will not be recognized for their contributions to successfully installed information systems. The source of ISD estimators' power is their knowledge of the details of the estimating process.
5. ISD implementers seek to install systems at, or below, the estimate. They may have no power in the creation of initial estimates. However, by completing projects within the estimates, they will achieve recognition among managers and users. in contrast, overruns give them very bad reputations.

These managers and staff members attempt to realize their objectives as they carry out the following systems development phases as predicted by the Political Model. The discussion further illustrates the conflicts of interest between these managers and staff, and the use of their relative power to resolve them.

Phase 1—The Initial Investigation

In this phase, user management or the user contact comes up with an idea and informally asks an ISD representative for a provisional estimate.

These requests have taken place during a planned meeting with the ISD representative, during a phone call, or even during a meeting of senior corporate executives. In one case, a senior executive briefly left a meeting with other top executives to find an ISD representative who would immediately provide an estimate to take back to the meeting. The ISD representative who provided the estimate had no previous knowledge of the proposed system.

Although these premature estimates have later proved inaccurate, the authors believe the ISD representative can increase his or her visibility (at least in the short-run) by providing it. The ISD representative can appear confident, intelligent, and authoritative. Furthermore, it is unlikely that the estimate will be challenged because no one else knows the scope of the project (according to the manager of Purchasing, a user) and because the application might never be developed (according to the procedures defined in the Standards and Guidelines Manual). Likewise, the ISD representative is often not the implementer; someone else might need to meet the estimate. Both the manager of the Systems Service Center and the manager of Systems Planning described this separation of responsibility in the interview. Hence, the authors conclude that it some cases, the ISD representative has spoken too freely with the knowledge that he or she will probably not be accountable for this estimate.

However, the authors believe if the ISD representative refused to give an estimate, then he might appear incompetent. By offering the estimate, the individual might seem very confident and very capable.

Unlike the Rational Model, which assumes that management and staff will disregard the Phase 1 estimate when better ones are developed, the Political Model recognizes that user management sometimes never forgets it, ISD management often views the Phase 1 estimate as imprecise. but user management often sees it as "cast in concrete." Hence, it may be the most important estimate. Later, if user management chooses, it can recall its inaccuracy to challenge the capability of the ISD and to discredit other ISD efforts. The manager of Financial Information Systems illustrated this when he said, "ISD's worst enemy is when management comes in and asks us to give a ballpark estimate during a meeting. The estimate goes into the minutes of the meeting. You will never replace such a ballpark estimate." Similar sentiments were expressed by the manager of Systems Planning and a programmer, while two users—the controller and the manager of Purchasing—stated that they do disregard initial estimates.

Phase 2—Preliminary System Study

During Phase 2, the ISD estimator officially examines the proposed system to develop a more formal estimate to justify the project. However, the Phase 1 estimate has in some cases already created a significant problem.

That is, at the beginning of Phase 2, the user contact has received the Phase 1 estimate but has still not identified system benefits. Hence, the contact can consider the Phase 1 cost estimate while claiming benefits, but in doing so, need not be completely candid. During the interview, the manager of Cost Accounting and Lease Accounting (a user) stated that the user contact can inflate predicted benefits in order to achieve project justification and not necessarily be caught. "ISD management needs to know the benefits of a project in financial terms but doesn't care whether the figure is correct... It's a game," he stated. Hence, the authors deduce that the project justification principle (i.e., benefits must exceed costs) is compromised by the benefits projection. The cost-benefits analysis presented in Phase 2 is sometimes fictional, and that estimating process might actually undermine project justification.

Moreover, if the user contact believes that the cost estimate will be too high (meaning it is impossible to reasonably fabricate offsetting benefits) yet wants the project authorized, then he or she sometimes reduces the project scope to increase its chance of acceptance. The ISD estimator lowers the estimate for the reduced project. However, the user contact may have privately planned to increase the scope in a later phase. The user contact finally does increase the scope and claims that the ISD estimator misunderstood the original requirements. For example, one interviewee, the manager of Marketing Information Systems, stated, "Users back off of their initial request to cut the estimate to get approval and then tack their earlier needs back on later to get what they had originally wanted."

The user contact has little difficulty carrying out this temporary scope reduction. The authors noted that the lack of an independent control or audit permits it. Also, because some benefits are intangible and their values are difficult to predict, the user contact can merely decrease their worth to pressure the ISD estimator to lower the cost estimate. A programmer/analyst and a contract programmer who were interviewed both described such pressure to cut their estimate, while the manager of Marketing Information Systems stated, "ISD management might cut an estimate to meet a budgetary constraint. This cut is done after a review with upper-level ISD management who deem an estimate too high for whatever reason no matter how good the estimate was. Perhaps an agreement had been made at the Director level, an informal high-level agreement over lunch." Hence, the ISD estimator acquieces if he very much wants to achieve his objective of gaining project authorization.

This lowering of the cost estimate by the ISD estimator is not difficult. The authors observed that two flaws in the method permit the ISD estimators to create estimates that fit their objectives.

First, the method lacks firm objective criteria for categorizing modules by level of complexity. Eleven of the 17 ISD interviewees discussed this lack of objectivity. There are no written, or otherwise consistent and objective guidelines for this determination. That is, the estimator intuitively identifies the complexity level of each module. The ISD estimator is the systems specialist most knowledgeable about the application and can lower the cost estimate by reducing the stated complexity level of the modules without being detected.

Second, the method assumes that all or most application functions are known. However, by necessity, the Phase 2 estimate is prepared too quickly and therefore without a deep enough understanding of the application and its requirements. The ISD estimator can rely on a limited understanding of the application functions to supply a lower estimate.

After the ISD estimator has worked with the user contact to prepare the Phase 2 estimate, it must be officially approved according to the Standards and Guidelines Manual. Two negotiations take place to gain approval. First, ISD management and the ISD estimator negotiate the estimate. Ten of the 17 ISD interviewees discussed this negotiation process. Generally, ISD management wants to reduce the estimate. The manager of Systems Development referred to cost estimates when he said, "They're too high and I want the job done quicker... whatever they come up with." The ISD estimator (particularly if he or she expects to be the implementer) wants to maintain it. A contract programmer admitted that he built 10 percent to 15 percent slack into his estimates to be prepared for unexpected changes, and added, "It is better to come in under than over. There is a little politics involved." Nevertheless, ISD management sometimes persuades the ISD estimator to reduce the estimate.

The rationale behind this possible reduction is an ISD management belief that by reducing an estimated cost, an actual cost can also be reduced. The manager of Systems Development illustrated this with his previous quote. Presumably, according to him, systems analysts and programmers will work harder if pressured to meet a lower estimate. Furthermore, ISD management sometimes suspects that the ISD estimator (especially if he or she will later be the implementer) may have already excessively padded the estimate in order to prevent an overrun. A senior systems specialist interviewee stated that his "little extra padding" was not excessive but that ISD management tried to remove it with an ensuing argument. Six ISD interviewees acknowledged the reduction of padding.

Thus, the authors concluded that if ISD management does believe an estimate is too high (where too high means that user management will not authorize the project), then it sometimes lowers the estimate to enhance the chance of acceptance. ISD management can do this because no one is likely to successfully protest. The estimate will be an established fact when the ISD implementer first sees it. Furthermore, the ISD implementer will not protest if ISD management has expressed that the failure to meet the estimate will not affect the next performance review. One programmer/analyst and a senior systems specialist stated that during performance evaluations, they were not accountable for meeting estimates, while the managers of Manufacturing and Engineering Information Systems, of Marketing Information Systems, and of Financial Information Systems stated that they held their subordinates accountable.

The second negotiation is between ISD management and user management. User management usually wants a lower estimate (implying a lower final cost), while ISD management usually wants to maintain the level of the estimate to prevent an overrun. The manager of Engineering Systems (a user) illustrated the former point when he suggested that his department did not want to "overspend," while the manager of Systems and Programming Support illustrated the latter point by stating that ISD was concerned about its perceived accuracy in estimating. This negotiation is limited because user management cannot negotiate on the basis of any understanding of the estimating process. The manager of Cost and Lease Accounting, the manager of Purchasing, and the manager of Engineering Systems (all users) confirmed that they did not understand the means whereby ISD arrived at its estimates. Nevertheless, the negotiation sometimes forces both parties into inflexible positions. it can force user management to threaten project cancellation if the estimate is now lowered, as illustrated by the threat of the cancellation of an imports order entry project. Conversely, it can prevent ISD management from lowering its estimate without revealing that the estimate was based on subjectivity and a lack of understanding of the application and requirements. For instance, ISD could not sufficiently lower the imports order entry project estimate because it could not explain the rationale behind the original estimate.

Sign-off completes the Phase 2 estimate approval process. Its purpose might appear to ensure an accurate estimate. However, the signatures on the Phase 2 estimate neither ensure the accuracy of the estimate nor approve its correctness. instead, user management signs to indicate a willingness to pay the estimate. According to the manager of Engineering Systems (a user), "Sign-off really does not mean much." ISD management signs to indicate that the ISD will develop the system. The manager of Systems Development stated that there were enough other signatures on the document that he feels "fairly comfortable," while the manager of Marketing Information Systems stated that her signature reflects only an acceptance of the estimate because she "rarely gets directly involved in estimating." Only the estimator's signature affirms the estimate. That signature indicates that someone (probably someone else) can complete the project within the estimate.

Phase 3—System Planning

Phase 3 includes a thorough analysis of the system with extensive user participation, and based on it, a possible revision of the estimate. According to the manager of Systems Planning at the beginning of this phase, the estimator gives the documentation to the implementer and withdraws from the project. The implementer learns more about the proposed system during this analysis.

The ISD implementer usually prepares an estimate by applying the rough/refine technique at this time. When doing so, the implementer often discovers that the application requirements were more complex than previously known. For instance, during the interview, the controller (a user) described a stock transfer project in which such a discovery took place. Hence, the implementer sometimes wants to increase the number and complexity of programs to raise the estimate and lower the chances of overrun. A senior systems specialist stated that he had to put as much padding in Phase 3 as possible because more explanation would be required if he tried to pad in Phase 4.

Usually, however, the ISD implementer declines to change the estimate. The influence of the Phase 2 estimate is substantial. ISD management sometimes forces the implementer to keep it. A programmer/analyst stated in the interview that she had no idea how her manager had arrived at Phase 1 and Phase 2 estimates and "felt forced to fit her estimate into the previous estimates." Alternatively, the ISD implementer sometimes chooses to prevent any embarrassment to the ISD that a change can cause. Another programmer/analyst described a system originally estimated at 220 days, but later re-estimated at 340 days; however, the original number was used to set up a target date and prevent any embarrassment in the eyes of the users.

Moreover, the implementer sometimes does not feel confident contradicting the Phase 2 estimate. This is because the implementer assumes that the ISD estimator better understood the system during Phase 2 than he or she (the ISD implementer) does in Phase 3. One programmer/analyst expressed lack of confidence doing so because Phase 1 and 2 estimates were usually done by more experienced senior systems specialists or planners.

Thus, the manager of Marketing Information Systems stated that of the seven or eight major systems she had seen estimated, only the estimate of one had been officially changed.

Nevertheless, the user contact's requirements sometime change. The user contact learns more about the potential of a new system. The manager of Systems and Programming Support described how both the users and ISD recognized the growth of the users' requirements during the development of a material requirements planning system. The user's needs (or understanding of those needs) grow, and hence the system requirements grow. Both the user contact and the ISD implementer usually recognize these new needs.

Thus, the ISD implementer wants to increase the estimate to reflect the new requirements and then have the user contact sign off on the new Phase 3 estimate. However, the user contact sometimes wants to maintain the Phase 2 estimate because an increase could imply his or her incompetence during Phase 2. The manager of Cost and Lease Accounting (a user) pointed out that because users for whom he served as liaison depended on timely project completion, changes to target dates were an embarrassment for him.

To prevent the ISD implementer from raising the competency issue, the user contact sometimes suggests that the ISD estimator or ISD implementer failed to understand the organization's needs and also failed to explain all of the system's possibilities earlier in the development process. Many interviewees admitted that the failure to understand user requirements was a major impediment to accurate estimates. For example, the controller (a user) stated, "ISD and Accounting initially thought that they understood a written document describing a project but later learned that they did not. They had different interpretations." Thus, to avoid criticisms, the ISD implementer simply chooses to keep the Phase 2 estimate, when in fact, it should be increased. A senior systems specialist commented that "lots of explaining is necessary if there is a change of greater than 20 percent."

Phase 4—System Design

Phase 4 represents the preparation of a detailed system design. In terms of estimating activities, the implementer uses rough/refine once again, but with a considerably better understanding of the proposed system.

Conflict between the ISD implementer and user contact at times occurs as in Phase 3. The resulting Phase 4 estimate differs from the Phase 3 estimate.

Phase 5—Program Specifications

Generally, a new estimate is not prepared during this phase because of limited user contact.

Phase 6—Programming

ISD has the best handle on estimating during Phase 6, according to a senior systems specialist that was interviewed. Thus, some estimates have changed during this phase. When this happens, the ISD implementer must obtain a sign-off on a new estimate or must attempt to meet the previous estimate to avoid an overrun.

The ISD implementer uses three basic tactics to meet the estimate (i.e., substantiate its accuracy) and avoid an overrun. The first tactic is to defer deliverables. That is, the ISD implementer identifies a portion of the system and promises to complete it on schedule, while delaying the completion of the remainder. In this way, the ISD implementer can argue that most of the system has been completed within the estimate. The manager of Systems Planning illustrated this by saying, "It sometimes ends up that you do what you can for the amount of money approved. This budgetary limit sometimes translated into deferred items."

The ISD implementer's second tactic is to work harder and faster. Since systems development is largely mental work, ISD implementers are able to work harder and faster when under pressure. "Crashing the project" sometimes brings it in within its estimate. For example, the manager of Systems and Programming Support stated, "If a system overruns, ISD works overtime so that the user does not know about the overrun." A programmer/analyst added, "Meeting a completion date is the real issue. You have to work overtime to meet the date if there is insufficient staff."

The ISD implementer's third tactic is to refrain from recording hours worked on the project or to assign them to other tasks. The absence of tracking of overtime hours makes this especially easy to accomplish. Hence, "creative accounting" permits the ISD implementer to appear to meet the estimate. As the manager of Systems and Programming Support pointed out, "ISD never tells the user about a mistake. They would rather assign additional resources not charged to the user."

Phase 7—Implementation Planning; Phase 8—Testing

In terms of recognizing overruns, these phases resemble Phase 6.

Phase 9—User Training

There are no estimating activities in this phase.

Phase 10—Implementation

At cutover, the ISD implementer has one final tactic to create the appearance of having met the estimate. He or she can flatly claim that the project was completed within the estimate even if that appears not to have been the case. The ISD implimenter can declare that the user contact drastically altered the requirements during development and that any apparent overrun was attributable to this. Hence, the ISD implementer can assert that the estimate was accurate and the original project was, or would have been, completed within the estimate. The manager of Systems and Programming Support illustrated this belief when he stated that "the major problem is the perception of inaccuracy" because of significant changes in user requirements rather than because of inaccurate estimating per se.

Another View of the Two Perspectives

While the evidence for the Rational and Political Models came primarily from the interviewers' request that subjects explain how costs were estimated, the answers to the request to identify any potential causes of estimating problems can be interpreted similarly (as can the answers to the request to identify the potential means for improving the process—suggestions that were complementary to the causes and are therefore not shown). Table 2 shows that 10 most frequently mentioned causes of the problem. The reader may easily interpret them as legitimate causes of inaccurate estimating in terms of the Rational Model where the process has an agreed upon and stated goal of providing an accurate estimate. However, the problems they represent may also be interpreted under the Political Model as opportunities for pursuing actions driven by personal motives. Explanations for each item in Table 2 convey this interpretation.

Summary and Conclusions

User and ISD management demand much of an information system cost estimate. An estimate initially helps to justify each individual proposal and thus helps to establish a portfolio of applications. Moreover, the estimate can also serve to schedule the development of the application, to staff it, to control and monitor its progress, to evaluate estimators and implementers, and even to market a system to users or to convince the ISD to develop a system. Both of the models described in this report illustrate those uses of the estimate and also provide additional implications for managers and researchers.

Implications for Managers

The study's most important implication to ISD managers is the formal recognition of the existence and pervasiveness of the Political Model. At the authors' presentation of their findings to the ISD director and other top ISD managers, some attendees observed that while they had been aware of much of their organization's politics, they had never before realized there might be so many examples of it in a single activity. This demonstrated its potential pervasiveness and its possible deleterious effects. Moreover, it suggested that as ISD managers, they should be even more sensitive to it than they had been and that they should be prepared for action to prevent or respond to it.

At that presentation, the authors provided the chemicals manufacturer with detailed recommendations to reduce the uncertainty inherent in the estimating process and thereby improve its accuracy. These recommendations were inspired by both the Rational and Political Models and are consistent with the management literature. They are also useful to other organizations. The recommendations are:

Educating: A course for inexperienced users to help them understand the importance of explaining their needs to estimators and a course for ISD estimators to help them learn to better elicit user needs. Both of those recommendations reduce the uncertainty of the estimating process as imposed by unclear requirements. They are consistent with Gailbraith's[40] observation that the increased

Table 2. Potential Causes of Estimating Problems

1. Inability to anticipate programmer characteristics (10)	The estimate considers the skill level of the eventual programmer. However, the identity of the programmer is not known when the estimate is prepared. This uncertainty will enable the estimator to increase the estimate later or to explain an overrun by claiming that he or she expected more capable programmers. Thus he or she avoids culpability for the overrun.
2. Sloppy estimating with overlooked tasks (8)	The estimator can rely on his or her lack of understanding of the application (with the resulting overlooked tasks) to justify an increase in the estimate. The estimator may claim that the lack of understanding arose from insufficient explanations from the user.
3. Insufficient user/analyst communication and understanding (7)	As in #2, both estimator and user can claim that the other's failure to communicate resulted in the need to increase the estimate.
4. Lack of methodology or guidelines (5)	The lack of objective guidelines for identifying the level of complexity of a program gives the estimator latitude in assigning the estimate with little challenge.
5. Lack of coordination of ISD functions (5)	Each of the ISD areas, such as Planning, Systems Development, Database Administration, and Operations, have their own needs and priorities. Delays in one area affect the schedules of many projects. Late in the development cycle, the estimator attributes the need for a new estimate or for an overrun to the failure of other areas to deliver their resources when needed.
6. Management removal of padding (5)	The estimator seeks a high estimate to prevent an overrun while management seeks a lower estimate in order to provide less costly service.
7. Lack of setting and review of standards (4)	The lack of agreed-upon durations for programs of each complexity gives the estimator flexibility in assigning the estimate.
8. Lack of historical data (4)	Similar to #7, the lack of well-kept records of previous projects gives the estimator flexibility in assigning the estimate.
9. Frequent user change requests (4)	The implementer can attribute the need to change the estimate to new user requirements.
10. Changes in ISD personnel (4)	The estimator can attribute the need to change the estimate to errors by personnel formerly associated with the project.

Note: Number in parentheses denotes number of interviewees who mentioned that cause.

participation of lower-level organizational members in decision making can help reduce task uncertainty.

Marketing: Executive briefings, white papers, and newsletters encourage user commitment, help dispel the perception that all ISD projects will overrun their estimates, and encourage mutual adjustment and communications. These recommendations represent Thompson's[41] assertion that as organizational units move from pooled interdependence to sequential or reciprocal interdependence, coordination is best achieved by mutual adjustment and communications.

Managing: Procedures for reducing conflicts such as the implementation of a new project management and control system, the implementation of an internal EDP auditing function, the increased involvement of implementers in the estimation process, the prohibition of the premature quoting of an estimate, the rewarding of project completion within the estimate to serve as an incentive to estimators and implementers, and the requirement for a more detailed statement of benefits before cost estimating. The new project management and control system and the rewards for timely project completion illustrate Ouchi's[42] view that linking performance (accurate estimates) to incentives is valuable when outputs (the accuracy of estimates) are easily monitored, but a transformation process (the recreating of the estimate) is difficult to monitor.

Tracking and controlling: Implementation of a detailed, standardized and periodical progress reporting system, the development of a new comprehensive checklist of potential system development tasks, and a more thorough collection of historical data. These procedures illustrate Cyert and March's[43] suggestion that uncertainty can be avoided by imposing standard operating procedures and industry tradition.

Auditing: A more careful check of delivered systems against needs, of actuals against estimates, of user-claimed benefits, of the progress of project development, of the adherence to change request sign-off, and of the accuracy of project control system reports.

Estimating: Improvement of the existing rough/refine methodology or the acquisition of a cost-estimating software package.

By attempting to reconcile the two models, these recommendations could lead to estimates closer to their ultimate, actual costs. The marketing and educating recommendations diminish the deleterious effects of the Political Model by establishing constructive attitudes that emphasize cooperation and candor. The managing, tracking and controlling, and auditing recommendations reduce the detrimental effects of the Political Model by checking and restraining counterproductive, motive-driven behavior. The managing, tracking and controlling, auditing, and estimating recommendations anticipate and prevent many sincere oversights resulting from the Rational Model. Hence, the process would be improved by being left neither entirely to the whims of the politics of the organization nor to the shortcomings of the managers, estimators, implementers, and users.

The ISD director responded to these recommendations at the presentation by stating his intent to implement a new project management and control system along with a new project management and cost-estimating training program for ISD estimators and implementers. He also said that he and his managers would consider other recommendations. Several of the interviewees stated that the study made them more aware of the problems of cost estimating. Each had experienced some of the conflicts described in the study, but none had previously recognized their widespread impact on each other.

However, many of the recommendations represented politically sensitive and time-consuming activities. Several of them would limit the flexibility of the managers responsible for their approval. Others would demand more time from already busy users. Moreover, the sponsor of the study and its champion, the ISD director, retired a few months later, during a shake-up of other top ISD managers. Hence, a year and a half after the study, the chemicals manufacturer had followed merely one recommendation—the trial acquisition of a new estimating software package—but was still strongly considering a new project management and control system, as well as several other recommendations. The delay in management action was probably attributed in large part to the ISD management changes.

Implications for Researchers

This research has provided an independent case study validation of Robeyand Markus' (1984) proposed Rational and Political Models of IS development[27]. The research extends their initial, sparse, and illustrative examples of the models into an in-depth analysis of a specific, critical dimension of IS development. No research had previously attempted to validate and extend those models in such a fashion. Moreover, the study offers the following implications for future researchers to consider in their investigations of cost estimating.

The Rational Model illustrates that the creation of an estimate is not a single event, but is instead, a multiple, serial process of considerable effort to continuously create, refine, and perfect a number. This suggests that when researchers objectively study algorithmic estimating methods, they need to record the actual estimates from each development phase to compare to the actual, final project duration.

The Rational Model also illustrates the importance of an estimate early in the development process. User and ISD management need the estimate well before ISD estimators can thoroughly understand the requirements of the proposed information system. Hence, methods that can provide early estimates are particularly valuable and are thus an important target for research.

In contrast, the Political Model reveals an estimation process with perhaps greater resemblance to a game of tug-of-war than to a prescribed management process. It demonstrates that participants in the system development process may be driven more by personal objectives than by organizational goals. Their fear of overrun and project cancellation may carry considerable influence. While this case research has revealed the existence of some error in cost estimating explainable by the Political Model, future research should assess the impact of that error. It may be significant.

Moreover, the research implications of the Political Model contradicts those of the Rational Model in an unexpected way. While the Rational Model suggests the use of objective studies comparing actual durations with the actual estimates from each phase, the Political Model disputes the accuracy of the actuals. Indeed, the "creative accounting" of project durations may invalidate such research. Researchers should investigate the extensiveness of "creative accounting" and should take care to ensure that estimates and actuals used in their research are as accurate as possible.

Most important, the Political Model demonstrates that the participants can attempt to subvert the Rational Model in order to achieve their own objectives. The possibility of their success raises a question mentioned earlier: Is cost estimating a management concern because of ineffective techniques or because of the poor planning and control of the estimating and development process?

Undoubtedly, the proponents of estimating techniques would argue in favor of the logic and elegance of their methods and would assert that managers simply must control their subordinates. Conversely, managers might argue that the methods fail to provide an accurate estimate early enough in the development process. In addition, ISD managers might argue that the methods' input parameters are easily compromised because they rely too heavily on the expertise and subjectivity of the user contact, ISD estimator, and ISD implementer.

This report has not definitively answered the question of whether poor estimating techniques or poor planning and control are responsible for making cost estimating a management concern. Still, by focussing on a management point of view and by considering the Rational and Political Models, it has provided a new perspective on cost estimating for further debate and investigation.

However, a single case study might not accurately represent cost estimating in all other firms. Therefore, with this case as the point of departure, researchers should study the management of cost estimating with a larger sample for the testing of hypotheses. They should attempt to ascertain the actual state of practice in cost estimating with such a sample. Finally, they should attempt to determine the extent to which the observations in this case study hold true and the extent to which they actually affect the process of information management.

References

[1]Emery, J.C. Cost/Benefit Analysis of Information Systems, The Society for Management Information Systems, Chicago, IL, 1971, pp. 16-46.
[2]King, J.L. and Schrems, E.L. "Cost-Benefit Analysis in Information Systems Development and Operation," Computing Surveys (10:1), March 1978, pp. 19-34.
[3]Guimaraes, T. and Paxton, W.E. "Impact of Financial Analysis Methods on Project Selection," Journal of Systems Management, February 1984, pp. 18-22.
[4]Drury, D.H. "An Evaluation of Data Processing Steering Committees", MIS Quarterly (8:4), December 1984, pp. 257-265.
[5]McKeen, J.D. and Guimaraes, T. "Selecting MIS Projects by Steering Committee," Communications of the ACM (28:12), 1985, pp. 1344-1352.
[6]BusinessWeek. "It's Late, Costly, Incompetent—But Try Finding a Computer System," Issue 3078, November 7, 1988, pp. 164-165.
[7]Allen, B. "An Unmanaged Computer System Can Stop You Dead," Harvard Business Review, November-December 1982, pp. 77-87.
[8]Lederer, A.L. and Mendelow, A.L. "Issues in Information Systems Planning," Information and Management (10:5), May 1986, pp. 245-249.
[9]Boehm, B.W. "Software Engineering Economics," IEEE Transactions on Software Engineering, January 1984, pp. 4-21.
[10]Freiman, F.R. and Park, R.D. "PRICE Software Model-Version 3: An Overview," Proceedings of IEEE—PINY Workshop on Quantitative Software Models, IEEE Cat. TH0067-9, Kiamesha, NY, October 1979, pp. 32-41.
[11]Herd, J.R., Postak, J.N., Russell, W.E. and Stuart, K.R. Software Cost Estimation Study—Study Results, Final Technical Report RADC TR-77-220, Vol. 1 (of two), Doty Associates, Inc., Rockville, MD, June 1977.
[12]Jensen, R.W. "An Improved Macrolevel Software Development Resource Estimation Model," Proceedings of the 5th ISPA Conference, St. Louis, MO, April 1983, pp. 384-389.
[13]Nelson, E.A. Management Handbook for the Estimation of Computer Programming Costs, AD A648750, System Development Corporation, Santa Monica, CA, October 31, 1966.
[14]Putnam, L.H. "A General Empirical Solution to the Macro Software Sizing and Estimating Problem," IEEE Transactions on Software Engineering, July 1978, pp. 345-361.
[15]Walston, C.E. and Felix, C.P. "A Method of Programming Measurement and Estimation," IBM Systems Journal (16:1), 1977, pp. 54-73.
[16]Wolverton, R.W. "The Cost of Development Large-Scale Software," IEEE Transactions on Computers, June 1974, pp. 615-636.
[17]Albrecht, A.J. "Measuring Application Development Productivity," GUIDE/SHARE Application Development Symposium Proceedings, Monterey, CA, October 1979, pp. 83-92.
[18]Donelson, W.S. "Project Planning and Control," Datamation, June 1976, pp. 73-80.
[19]Halstead, M.H. Elements of Software Science, Elsevier North Holland, New York, NY, 1977.

[20]McCabe, T.J. "A Complexity Measure," IEEE Transactions on Software Engineering (Vol. SE-2), December 1976, pp. 308-320.
[21]Computer Associates. CA-ESTIMACS: An Applications Development Project Estimation System, Mt. Laurel, NJ, 1987.
[22]Kemerer, C. "An Empirical Validation of Software Cost Estimation Models," Communications of the ACM (30:5), May 1987, pp. 416-429.
[23]Benbasat, I. and Vessey, I. "Programmer and Analyst Time/Cost Estimation," MIS Quarterly (4:2), June 1980, pp. 30-43.
[24]Benbasat, I., Goldstein, D.K. and Mead, M. "The Case Research Strategy in Studies of Information Systems," MIS Quarterly (11:3), September 1987, pp. 369-388.
[25]Roethlisberger, F.J. The Elusive Phenomena, Harvard Business School, Division of Research, Boston, MA, 1977.
[26]Bonoma, T.V. "A Case Study in Case Research: Marketing Implementation," Working Paper 9-585-142, Harvard University Graduate School of Business Administration, Boston, MA, May 1983.
[27]Robey, D. and Markus, M.L. "Rituals in Information System Design," MIS Quarterly (8:1), March 1984, pp. 5-15.
[28]Jones, C. Programming Productivity McGraw-Hill, New York, NY, 1986.
[29]Keen, P.G.W. "Information Systems and Organizational Change," Communications of the ACM (24:1), January 1981, pp. 24-33.
[30]Pfeffer, J. Power in Organizations, Pitman Publishing, Inc., Marshfield, MA, 1981.
[31]Wildavsky, A. The Politics of the Budgeting Process, third edition, Little, Brown and Company, Boston, MA, 1979.
[32]Pfeffer, J. Organizational Design, Ahm Publishing Corp., Arlington Heights, IL, 1978.
[33]Allison, G.J. Essence of Decision, Little Brown and Co., Boston, MA, 1971.
[34]Borun, F. "A Power-Strategy Alternative to Organization Development," Organization Studies (1:2), 1980, pp. 123-146.
[35]Grover, V., Lederer, A.L. and Sabherwal, R. "Recognizing the Politics of MIS," Information and Management (14:3), March 1988, pp. 145-156.
[36]Lund, R. "Indirect Participation, Influence, and Power: Some Danish Experiences," Organization Studies (1:2), 1980, pp. 147-160.
[37]Thompson,J.D. and Tuden, A. "Strategies, Structures and Processes of Organizational Decision," in Comparative Studies in Administration, J.D. Thompson, P.B. Hammond, R.W. Hawkes, B.H. Junkerand A. Tuden (eds.), University of Pittsburgh Press, Pittsburgh, PA, 1959, pp. 195-216.
[38]Lucas, H.C. Jr. Information Systems Concepts for Management, third edition, McGraw-Hill, New York, NY, 1986.
[39]Bardach, E. The Implementation Game: What Happens After a Bill Becomes a Law, MIT Press, Cambridge, MA, 1977.
[40]Galbraith, J.G. Organization Design, Addison Wesley, Reading, MA, 1977.
[41]Thompson, J.D. Organizations in Action, McGraw-Hill, New York, NY, 1967.
[42]Ouchi, W.G. "A Consensual Framework for the Design of Organizational Control Mechanisms," Management Sciences (25:9), September 1979, pp. 833-848.
[43]Cyert, R.M. and March, J.G.A Behavioral Theory of the Firm, Prentice-Hall, Inc., Englewood Cliffs, NJ, 1963. ■

Cooperative Processing: Its Organizational Impact

In this report:

This report will help you to:

- Gain a management perspective on the distributed processing architecture.
- Gauge distributed processing's impact on your organization.
- Determine which organizational area will be most affected by distributing processing, and how best to manage the conversion.

For the past few years, people in MIS departments and the PC advocates of end-user departments have been locked in a feud. The basic disagreement often comes down to the often-heard but poorly understood question, "Who owns the data?"

MIS people maintain that they own the data. They control access to the organization's data because it is on their systems and, in most cases, it is accessed through applications they have developed and maintained. At the same time, the people from user departments feel they own the data. It is, after all, data about their departments, their products or services, and their employees. They enter it. They maintain it. They use it.

There is no right or wrong answer to this question, and every organization must come to its own conclusions. And there is probably no philosophical answer that will satisfy everyone. There is, however, a technical answer that holds some promise, and that is cooperative processing.

A Definition of Cooperative Processing

Cooperative processing is an application architecture or topology allowing two or more computers to share the processing of a program. This admittedly broad definition could describe applications developed using several different types of programmer tools, communications

---

This Datapro report is a reprint of "Are You Ready for Cooperative Processing?" by Ross Altman pp. 20-31, from *Information Center*, Vol. VI, No. 4, April 1990. Copyright © 1990 by Weingarten Publications, Inc. Reprinted with permission.

protocols, and computers. The accompanying sidebar provides more detail on the actual technology.

Regardless of specific approach, cooperative processing provides two significant benefits. First, it generally supports graphical user interfaces such as *Motif*, *Open Look*, *New Wave*, and Common User Access (CUA). Such interface standards, delivered via a PC, Unix workstation, or an X Windows terminal, provide the user with an improved application that is easily learned and easily used.

Second, cooperative processing can reduce the amount of processing performed on the computer that serves as the "host" or "server" for the system. This reduction is due to the division of tasks between the user's computer or terminal and the host computer. In a cooperative processing application, the host or server is used almost exclusively for data base I/O—for maintaining the application's main data base and for performing table lookups. This leaves the rest of the application executing on the user's PC or workstation. When compared to a conventional application, where all of the processing executes on the host computer, cooperative processing can reduce the processing load on the host by as much as 90 percent.

Examples

In the real world, cooperative processing has been used in almost every industry and for almost every business task. A container shipping company used it to provide an advanced PC-based interface to an executive support system. This application allows the company's managers to access mainframe reports without becoming computer experts or creating a significant drain on the time and resources of MIS.

In another cooperative processing application, a major consulting company with 55 worldwide offices needed to rewrite the application that they use to keep track of the specific skills and areas of expertise of each of their consultants. For a consulting company, this type of application is critical, since it is the equivalent of an inventory program for a manufacturing concern.

In the old system, changes to the consultant data base had to be submitted in writing to a data entry department in New York. This process meant that it took as long as 10 days to add the data on a new consultant to the system. Also, because the data on each consultant has to be transcribed twice (once onto a form and then to a data entry screen), there were twice as many opportunities for errors to occur.

With an easily learned PC-based interface, the cooperative processing application that they developed allowed the consulting company to move data entry responsibilities out of New York and into their remote offices. This improved the timeliness and the integrity of the consultant data base without imposing an unreasonably long learning process on the users in the 55 dispersed locations.

The commercial banking division of a major money-center bank provides another example. This bank wanted to provide its clients with a new customer service. It decided to develop an application with an easily used interface so an institutional client could directly access the bank's mainframe and review its account information.

Since the users of this application would be the vice president of finance or the controller of each client company, a typical dumb terminal-based application interface would be unworkable. Such an interface would be too difficult to learn and too cumbersome to use. Instead, the interface had to be based on the human factors principles incorporated into such standards as the Macintosh interface or IBM's CUA. So the bank's developers created a system with a PC interface for the clients and only uses the mainframe as a data base machine.

A company that allows customers to directly access their own account data on its computer can have a profound impact on its own competitiveness. Clients with a direct link to the company's mainframe-based data will feel they are getting better service. In an increasingly service-oriented marketplace, this can be a critical advantage. And it can become a critical disadvantage if your competitors provide this type of on-line access and you don't.

Although some applications based on conventional processing have been placed in the hands of end users and customers over the past few years, increasingly, cooperative processing is used for this purpose. The reasons for this shift in application architecture are quite clear. Users of these systems are mostly business people—purchasing agents, accountants, money managers, and so on—not computer operators. If the system cannot be learned and used productively in a few minutes it will probably not be used.

Who Owns the Data?

While cooperative processing is a superb architecture for many new applications, its rapid acceptance is not necessarily assured. Unfortunately, the decision concerning how an application should be developed is not always based on the objective criteria discussed above. More often than not, subjective and political issues are also critical considerations in these decisions. And the issue of who owns the data is probably among the most important.

For the moment, the argument over who owns the data has produced a standoff. In many companies, MIS continues to exert primary control over centrally stored data. Meanwhile, PC users are developing their own applications and downsizing others so they can liberate their data from the host.

The problem is that both sides have legitimate concerns that must be addressed by any proposed resolution. For example, the end-user departments often believe that MIS has been slow to develop new applications and slow to update existing programs. They may also believe that the applications MIS develops are based on conventional technology that does not provide the application interface they have come to expect, a graphical interface based on prevailing PC technology.

For its part, the MIS department has a legitimate set of concerns as well. It has, or feels it should have, responsibility for maintaining the integrity of the data gathered by the organization's computer programs, regardless of whether those programs run on PCs or mainframes. The MIS department is also acutely aware of the many ways that this data integrity can be compromised, such as concurrent updates of records by more than one user, irregular or incomplete backups, and insufficient data editing.

Add to this list that most data must be accessed by users in multiple dispersed locations, and you can see why the MIS department becomes concerned when it sees data stored on individual PCs or dispersed LANs. Backups are often omitted or performed using hardware, software, and procedures that have earned an unfortunate reputation for unreliability. PC-level data bases and LAN operating systems have immature provisions for journaling, dynamic transaction backout, and other basic capabilities of transaction processing. And most PC-based solutions to dispersed data access involve extensive duplication of data.

The arguments about who owns the data, though, do not usually address issues of efficiency, security, and integrity. They address emotional issues having to do with ownership and control.

In fact, the top managers in most organizations would maintain that *neither* the end users nor the MIS department owns the data. Quite the contrary. The *organization* owns the data, and it is the responsibility of the organization and its employees to manage that data in the most effective, efficient, and responsible manner possible. As it happens, the host is most often the best location for the storage of the organization's essential data, once the politics of the question are put aside.

By placing the data handling functions on the host, for example, the developer can ensure the highest level of security for the application's data, because the host data base is protected by the host security system. Host security systems are more mature than their PC counterparts and therefore do a better job of securing vital data from unauthorized access and manipulation.

By using the existing host data base, the developer of a new application can access data that is gathered by existing applications. Placing all of the new application's data on a mini or a PC data base server, on the other hand, could limit that application to the data gathered through the application itself. This would restrict the scope of the application to local data. As an alternative, the developer could design the application to work from a data extract, taken on a daily or weekly basis. This approach, however, would force the users to use data that may not be up to date, unless real-time data could be maintained in two machines at the same time. Software that supports this kind of configuration for transaction processing is not yet commonly available.

The host data base will also ensure that only one user will update a record at a time. And if the transaction is interrupted before the entire unit of work has been completed, then the partially completed transaction may be automatically backed ▶

The Different Styles of Cooperative Processing

There are a number of ways to share a program between two or more computers.

Front-End Processing
A PC program can be written to run in front of an existing host application without changing any code on the host application. This is done by writing PC code that issues calls to a PC-resident application program interface (API). The most commonly used API for 3270-based communications is the IBM High-Level Language API (HLLAPI—pronounced ha-LA-pee). There are actually five or so IBM versions of the HLLAPI, but all of them share a basic set of functions that permit a PC program to read data from the 3270 screen image, read the status line to check for the presence of the "X indicator," write data to the 3270 screen image, and then send a 3270 keystroke (e.g., a PF key, a PA key, Enter, Reset).

In front-end processing, the host application runs as before, sending screen images to what it believes to be a dumb terminal. On the user end of the 3270 connection, the PC program reads the host screen images by issuing calls to the API. The PC program then maps the data from the 3270 map's fields into its own fields. The PC program is then used to support the user interface and edit the data this is input by the user. When the user has completed working with the record, the PC program moves the data from the PC fields back into the 3270 fields and then asks the 3270 emulation program to send the map back to the mainframe.

While front-end processing offers the obvious benefit of supporting a new, PC-based interface without requiring changes to the host code, there are some who take an academic's approach to the subject and maintain that this isn't really cooperative processing. Others might reply that if it walks like a duck and sounds like a duck . . . .

Peer-to-Peer Processing
In a peer-to-peer processing application, two processors typically share the load of executing a program. Most of the time, one of the processors handles the task of presenting the interface to the user, while the other processor handles a more processing-intensive or I/O-intensive task, such as data base maintenance. Peer-to-peer processing is also used to off-load a computationally intensive task to a processor that is optimized for such work, perhaps with a vector processing facility, parallel processing architecture, or a math coprocessor.

Depending on the environment and the nature of the peers, peer-to-peer processing is referred to as LU6.2/APPC, remote procedure call (RPC), or client-server architecture.

LU6.2/APPC
In commercial applications, many computers executing peer-to-peer processing applications communicate with each other using IBM's SNA Logical Unit 6.2 and APPC or CPI-C. In this case, LU6.2 is the low-level communications protocol, while APPC and CPI-C are the APIs. APPC has been available for a number of years in functionally similar but syntactically different implementations for almost all IBM environments (the major exceptions are TSO and IMS/DC). Because of these syntactic differences, code written for one host APPC implementation is not portable to another APPC implementation.

In IBM's SAA, APPC has been supplanted with CPI-C. CPI-C itself is functionally similar to APPC, but it is not syntactically identical. So applications written under APPC today will have to be rewritten when they eventually migrate to CPI-C.

Remote Procedure Call
In the Unix and TPC/IP network worlds, RPC is used to implement peer-to-peer processing. Basically, RPC allows a program on one computer to call a subroutine that executes on another computer on the network. The code that is written for both the main program and the subroutine is exactly the same as it would be if the two pieces of code were on the same computer. The RPC tools provided by the network or operating system vendor take care of the communications work required to pass the procedure call to the computer that will execute the subroutine.

The primary difference between the RPC application and the more conventional peer-to-peer application is in synchronization. With conventional peer-to-peer applications, two programs run on separate computers. When one of these programs requires data or processing from the other computer, it opens a communication session and begins a conversation with the other program. At some point during the conversation, the program that initiated the link may go off and perform some other processing while the called program performs its assigned function. In effect, a peer-to-peer application implements a loosely coupled form of parallel processing.

With an RPC call, the processing of the calling program and the remote procedure do not overlap in this way. As it would if it were calling a local procedure, the calling program waits for the subroutine to execute. When it receives the data or the return code to indicate the results of the subroutine's work, the calling program continues processing.

Client-Server Architecture

In the PC LAN world, client-server architecture is most often used to describe how the current group of data base servers operate. In the old days (pre-1988), programs used the file server implementation of the LAN operating system. They would submit requests to the LAN workstation's DOS redirector software. This workstation software would trap all file I/O requests and look at the drive letter for the requested data. If the drive letter indicated that the file was on the file server, then the LAN software would take the file from the server and transmit it to the workstation. The program at the workstation would then continue processing as if it were working with a local file. Clearly, this required a lot of data movement for what could be a request for a small amount of data.

Unlike this file server implementation, the new generation of data base server programs split the data I/O task into two parts. The workstation program issues requests to the data base server. A program on the server machine evaluates the request, retrieves the requested data, and only transmits back to the workstation the requested data.

While the term client-server architecture most often refers to this type of processing, there are some cases in which the task that runs on the server is not a data base program. For example, the processing-intensive task on the server may be a statistics program or an expert system. Such applications may be implemented with an RPC product, or they may be implemented by using lower level communications protocols, such as APPC, Netbios, IPX, or Named Pipes.

Distributed Data Base

Distributed data bases represent perhaps the most seductive cooperative processing technology. In a fully distributed data base, the programmer or user will be able to use the tools of the data base product on the local processor (PC or workstation) without knowing the physical location of any data. The data base product will be able to find the requested data and retrieve it, or update it, even if it is stored on multiple dispersed processors connected over disparate network links. That capability is called location independence.

In addition to location independence, a viable distributed data base should have no single point of failure. In other words, all data should be stored on more than one processor in case one of those computers breaks down. And the viability of each *logical* communications link between two nodes on a distributed data base's network should not be subject to a malfunction in a single *physical* link. There must be multiple communications paths so that, again, there is no single point of failure.

That's the ideal distributed data base. Distributed data base products now available fall somewhat short of that target. While there are a number of products that can perform a query against data that is stored in multiple computers over a limited number of network configurations, none provides update capabilities along with the full location independence that is the cornerstone of a true distributed data base.

Even when this requirement is met, there will be a critical trade-off that must be accepted in order to enjoy the benefits of a distributed data base. In order to protect against a single point of failure, multiple copies of both data repository files and index files must be stored in different locations. With these duplicate repositories and duplicate indices, every update or add to a distributed data base will require at least twice as much processor and network activity as would be required with a nondistributed data base that supports a single repository file and a single index file. This additional processor activity and network traffic will be necessary in order to update the dispersed duplicate copies of each data record and each index. It is not difficult to imagine a distributed data base configuration in which more processor time and network capacity are dedicated to maintaining the backup elements of the data base than are used in maintaining the primary copy of the data base.

out by the transaction processing facilities of the host operating system and teleprocessing monitor. Similar facilities to protect the integrity of the data are not yet commonly available for PC LAN data bases.

If the data is maintained at the host, then the host's backup procedures and uninterruptible power supply will help to maintain the integrity of the application's data, regardless of power outages or operator error. As late as the end of 1989, articles and columns were still being written to recount stories of the lack of reliable equipment, software, and procedures necessary to ensure the proper backup of LAN data.

All of these reasons to centralize the storage of corporate data go back to one central theme. The data gathered through an organization's applications becomes a vital asset of the organization and must be managed as such. Considering the importance of maintaining and protecting this critical asset, there must be a truly compelling reason to move the storage and maintenance of data away from the central host computer. For the majority of applications, this compelling reason does not exist.

Applications Development

All of this doesn't mean, however, that the very real needs of the end users can be overlooked. After all, the inability of most MIS departments to quickly deliver effective applications is what caused end-user activism in the first place. So the suggestion that the mainframe is the best repository for most applications should not be taken as a justification to maintain the status quo.

In the area of applications development, there are two primary concerns that plague end users. First, application backlogs have grown to multiyear proportions. Critical changes in the way business is conducted must be put off because adequate systems support is unavailable. The second concern is related to the gap between the users' functional requirements and the ability of MIS to deliver applications that meet those requirements.

The implications of the application backlog are not always clearly understood. The backlog is more than an inconvenience for end users. It also adversely affects the competitiveness of the organization. The difference between an application's net benefit and the amortization of its development expenses represents an actual cost to the organization. For example, an improvement in a customer service application could produce a cost savings of $250,000 per month. If you can realistically amortize the development cost of that improvement at $100,000 per month, then the cost of *not* developing the application is $150,000 per month. It does not take too many of these opportunities sitting in the MIS backlog to amount to a significant cost.

With that in mind, think about the impact cooperative processing can have on the speed of development in the typical organization. By placing the vast majority of the logic for an application on a PC instead of on a host, you can reduce the development effort by a considerable amount. Factor in the use of any of the very advanced fourth-generation development tools that are available for PC development, and the opportunities are apparent.

But end users are not only concerned about the speed of application development. They are also concerned about the quality of the applications that are developed. More specifically, they are concerned with the user interfaces of those applications.

As long as applications are developed and delivered on the mainframe using conventional processing techniques, the user interface for those applications is constrained by the limitations of dumb terminal technology. As soon as the platform used to support the user interface shifts from a dumb terminal to a PC or some other intelligent workstation, however, the quality of the applications that are developed increases by an order of magnitude.

And that is precisely what the end users are looking for. Most of the time, end users really don't care where their data is stored. Their requirements are generally expressed in much more functional terms. They simply want applications developed quickly and they want these applications to offer the ease-of-use that PC programs have led them to expect from their computers.

Impact on the Organization's Mission

Generally, the benefits of a cooperative processing system will be readily apparent even when the system is used by existing data entry people. As a rule, though, the organization will realize considerably greater benefits when cooperative processing applications are used to change the way the organization works. For example, cooperative processing can be used to place an application's data entry function in the hands of line people, completely removing data entry clerks from the loop. This permits a redeployment of personnel and offers the possibility of a substantial reduction in costs. Also, by eliminating the manual process of filling out forms and sending them into a central site for data entry, this work-flow change can provide more timely and more accurate data.

There are other benefits, harder to quantify perhaps, that can have an even greater impact, as when allowing customers to directly review the status of their accounts improves the quality of the service an organization offers them.

Staffing Patterns and Reporting Relationships

But these benefits don't come without some costs. Along with the expected improvements in effectiveness and profitability, changes in work flow and responsibility may exert a considerable impact on some of the relationships that exist between work groups in the organization. Organization development professionals have known for a long time that changing the responsibilities of individuals and groups can shift the balance of power and influence in an organization. And these shifts may not always be perceived as positive by the people and groups that are losing power.

For instance, in the case of the bank that provides direct customer access to account information, this could shift customer dealings away from the customer service group and to the account managers, shifting power and influence with it. The managers of the "losing" group, fearing for their jobs, could fight the new application, dig in their heels, and throw up as many obstacles as possible to hinder the smooth implementation of the new system. Or they could try to support the transition and look for new ways for their group to help the bank make money.

Beyond good corporate sportsmanship, there is another reason for the managers affected by cooperative processing applications to go with the flow: general competitiveness. If the bank doesn't control costs and improve services, perhaps a competing bank will. And then, as customers move to the bank providing better services at better rates, the bank with the recalcitrant managers will be taken over by a healthier bank and the recalcitrant managers may well find themselves out of work, anyway. Under the circumstances, wouldn't it be better to work toward improving the operation of the bank instead of hindering it?

The Challenge of Managing an Architecture

Of course, it is easy to discuss enlightened management practices when it's not your ox being gored. In the real world, it's never that easy. But if you don't manage change, it has a way of managing itself, and that rarely works out well.

So how can an organization predict the impact of cooperative processing? And how can this impact be channeled to benefit the organization's mission rather than the personal agendas of some of the group's best political players?

It would take a book to answer these questions completely, but I can tell you some things to look for.

First, the impact of any organizational change will depend on the structure of the organization. Is it rigid or flexible? Is it hierarchical or does it have a flat, matrix-type management structure? Flexible organizations encourage their members to find innovative ways to perform their tasks. They don't have rigid expectations of how each task will be performed, and they don't deal harshly with individuals who attempt to perform their roles in new and innovative ways. Rigid organizations define the roles of individuals specifically and firmly enforce role expectations. In a rigid organization, you just don't do things differently. Between the two extremes of rigid and flexible organizations is a spectrum of organization types.

Often, rigid organizations are organized in a hierarchical manner. The lines of authority are clearly defined and reporting relationships are direct and exclusive. In other words, a mid-level manager reports directly to a single manager and has exclusive and direct authority over a group of direct reports. This is in marked contrast to the matrix-management scheme, in which, for example, a product manager may have dotted-lined responsibility for R&D and marketing without direct supervisory responsibility over those areas.

While it might seem that matrix-management schemes are implemented in flexible organizations and rigid organizations are hierarchical, this isn't always the case. Nor is it correct to assume that change is easier to implement in a flexible, matrix-managed organization and more difficult to implement in a rigid, hierarchical one. It's more complex than that.

Flexible organizations are not only more lenient with individuals who find new ways to get their jobs done, they are also more likely to alter the structure of the organization when conditions change. They tend to hire and promote people who are most comfortable in a flexible setting and to encourage those who need more structure to "loosen up." A culture like this is often a hotbed of innovation and is likely to find many ways to use cooperative processing to the benefit of the organization's mission.

Rigid organizations, on the other hand, are less likely to readily accept the changes that are necessary to successfully implement a cooperative processing system and to take full advantage of the benefits that such a system may offer. Role expectations are more often cast in concrete in a rigid organization and reporting structures and group responsibilities are just as clearly defined and firmly enforced as individual roles. An organization like that is not likely to rearrange itself, for example, to eliminate redundant positions, so some of the potential benefits of a new system may remain just that—potential.

Now, if flexible organizations are more likely than rigid organizations to implement change, and if flexible organizations are often matrix-managed, does it follow that matrix organizations are more likely to successfully adopt cooperative processing than hierarchical organizations?

Not necessarily. If the flexibility-rigidity element is equal in two differently managed organizations, the hierarchical one is more likely to be successful at change management than the matrix organization. The reason has to do with the simplicity of the hierarchical organization.

In matrix organizations, each dotted-line relationship is a negotiated relationship based on the common understanding of the individuals involved concerning who is responsible for what functions and who gets his way if there is a non-negotiable disagreement. When reporting relationships change, these negotiated relationships must be rebuilt. That is a time-consuming process. And since people don't get along with everyone they work with, some of the new relationships may be harder to renegotiate than others. The entire process takes a fair amount of time and effort.

By comparison, hierarchical organizations, with their direct and exclusive reporting relationships, are more easily recast than complex matrix-managed organizations. Since the reporting relationships are more clearly defined, it's easier to define the new set of roles for everyone, and it's easier to follow through in order to make certain that the new way of operating is implemented properly.

Flexible-hierarchical organizations are more likely to implement cooperative processing systems in a manner that will maximize the benefits to the organization. Flexible-matrix organizations will also find ways to successfully implement the technology; it will just take them longer.

For rigid-hierarchical organizations, however, the outlook isn't quite so good. In these settings, any type of change is threatening and difficult to manage. In particular, the battle lines between PC advocates and MIS may be drawn more clearly in these organizations, especially when the MIS organization has a poor track record for delivering new applications and has resisted providing more than token support for PC users.

This is particularly unfortunate because cooperative processing may be the perfect tonic for the stalemate that has evolved between the PC and mainframe camps in rigid organizations. The use of cooperative processing not only gives each of the combatants what it wants, but it does so by matching technological responsibilities to organizational ones.

Neither PCs nor mainframes are going to go away any time soon, and the sooner organizations learn to make the best of both, the more they will succeed in their other goals. ■

Managing Change During Information Systems Technology Implementation

In this report:

Datapro Summary

Although it is a natural phenomenon to resist change, the process of automating an organization often results in an even stronger level of resistance among employees. This resistance can seriously impede the implementation of computer technology. The equity-implementation (E-I) model is a theory-based guideline to understand users' apprehension and resistance to change in a computer setting. It helps managers understand, and prepare for, the inevitable resistance to systems implementation.

Introduction

Successful implementation of modern technologies, innovations, and management science is critical for enhancing the productivity and the competitive position of an organization.[1,2] However, the successful implementation of modern computer-based automation technologies in organizations remains a challenge. Researchers have noted widespread failures in the implementation of information systems.[3-7] A number of different approaches and theoretical perspectives have been adopted in examining and explaining the sources of implementation problems.

Some MIS researchers view the quality of the process of implementation as a major determinant of the success of implementation.[8,9] This view recognizes systems implementation as a change process and systems designers as change agents.[8,10] In recognition of the need to manage change, different perspectives and approaches have been proposed for understanding and managing the change process. These include the planned-change approach proposed by Lewin[11] and Schein[12] and further elaborated by Kolb and Frohman,[13] the innovation-process approach,[14] and the socio-technical systems approach.[3,15,16]

MIS researchers have also viewed the dynamics of implementation as a political process, where the sequence and the direction of implementation can be explained in terms of the conflicting interests of different user groups. Different user groups are viewed as competing to increase their power, to control information,[17-20] to obtain a greater share of computing resources,[19,21-25] and to achieve preferred task allocation.[24] Markus[19] explains resistance to change and implementation difficulties primarily in terms of the conflict among users for increased power. She notes that the political perspective appears to be primarily applicable for systems cutting across multiple user departments.

Some other factors considered relevant by MIS researchers in determining users' acceptance and assessment of a system include ease of use and usefulness,[26] prior expectations,[9] user involvement,[27] and impact on work environment.[28,29] Ginzberg, et al.[30] considered various individual, organizational, and system characteristics in their proposed model of implementation. These characteristics include management

This Datapro report is a reprint of "A Model of Users' Perspective on Change: The Case of Information Systems Technology Implementation" by Kailash Joshi, pp. 229-242, from *MIS Quarterly*, Volume 15, Number 2, June 1991.

support, user decision style, user knowledge, user job characteristics, user confidence in system, user demographics, goal congruence, and user involvement.

These studies and perspectives provide many useful insights into implementation problems and extend our understanding of implementation issues. MIS researchers recognize user acceptance of systems as a major objective of systems implementation.[30] While there are various perspectives on resistance to change, there is consensus that understanding and explaining resistance to change is important.[31] These explanations, even if informal or implicit, guide the behavior of systems implementors.[19] MIS researchers recognize that better theories or models of user resistance would lead to better implementation strategies and desired implementation outcomes.

This report examines the issue of IS implementation and resistance to change from an equity theory perspective. Equity theory is a well-established theory whose predictions have been found to be applicable in nearly all social settings.[32] The importance of equity issues in an organizational context has been recognized in the literature.[33]

This report uses equity theory to develop an equity-implementation model that attempts to explain resistance to change. The model is based upon the premise that there is no fundamental resistance to every change. For example, individuals readily adopt changes such as a pay raise or promotion. It is proposed, therefore, that individuals attempt to *evaluate* most changes. Changes that are considered favorable should not be resisted and may even be welcomed. However, changes considered unfavorable are likely to be resisted. The model utilizes equity theory to identify the processes through which users may evaluate changes introduced by IS implementation to assess whether changes are favorable or unfavorable to them. The model provides a framework for integrating some of the previous MIS research studies. The model also suggests some guidelines for managing implementation. It should be noted that the focus of this report is on a specific implementation or change rather than the whole MIS environment.

Previous equity research in the MIS-area identified procedural fairness, distributive fairness, and reciprocal fairness as the main dimensions of equity in the overall MIS context.[34] Research has also demonstrated the influence of equity on user attitudes such as user information satisfaction. This research has also discovered that procedural fairness is the most important fairness issue for users, followed by distributive and reciprocal fairness.[35] Equity has been found to have the highest causal influence on the overall user information satisfaction (UIS) compared to the other known factors that are likely to influence UIS.[36] Equity theory has also been used to explain user behaviors ranging from unfair behavior (e.g., political behavior and user conflicts) and reactive dysfunctional behavior (e.g., non-usage and non-cooperation) to fair behavior, such as seeking of fair structural arrangements for the allocation of MIS resources (e.g., steering committees, user control over budgets, decentralization, etc.).[37]

The next section analyzes a user's view of changes (introduced by an implementation from an equity theory perspective) and presents the equity-implementation (E-I) model. The following section presents three examples from the MIS context to illustrate the application of the model. Next is a discussion of the implications of the E-I model along with some guidelines for managing the implementation process. The final section includes a discussion of contributions of the research, limitations, and directions for future research.

Equity-Implementation Model

Equity theory suggests that in every exchange relationship, individuals are constantly concerned about their inputs, outcomes, and the fairness of the exchange. Individuals are also constantly comparing themselves with others in their reference group to assess whether the relative gains are the same.[32,38,39] In such an environment, when a change is introduced it is likely to change the inputs and outcomes of individuals and other parties involved in the exchange. When the changes in inputs and outcomes are such that on the whole the individual perceives a decline in his or her net gains, or inequity compared to others, the individual is likely to be distressed. Equity theory suggests the greater the inequity or decline in net gain, the greater the resulting distress would be. Individuals who experience the distress of inequity or loss of equity are likely to resent the change and resist it by attempting to minimize their inputs and others' outcomes as well as by attempting to increase others' inputs. On the other hand, individuals who perceive an increase in their equity are likely to welcome the change.

An important step in developing the model is to identify the different evaluation processes a user may employ in determining a change in equity due to an implementation. Given that we are dealing with an organizational context, the task of identifying relevant equity evaluation processes can be guided by the likely comparisons within an organizational context. In the following discussion the organization is termed the employer and an affected individual is termed the user. The model in Table 1 identifies three levels of analysis that an individual may use in assessing the equity situation. These three levels of analysis are discussed below.

First Level of Analysis

At the first level of analysis, a user would evaluate the impact or likely impact of the implementation of a new system in terms of the resulting change in his/her outcomes and inputs as noted above. If the net gain due to change (i.e., Δ Outcomes − Δ Inputs) is positive, users would be favorably inclined toward the change. However, it the perception of the net gain is negative, users would view the change as unfavorable and resent it.

Changes in outcomes are defined as the perceived benefits or losses that the implementation of a system brings about for the user. Some examples of change in outcomes may be better working conditions, less tension, loss of seniority, or increased risk of losing one's job. Similarly, changes in inputs are the additional efforts, skills, or abilities that a user may need to bring to the job. Some examples of changes in inputs may be increased workload, increased cognitive effort, or fewer physical tasks. Table 2 lists some likely changes in the outcomes and inputs for users due to an information system or technology implementation.

Second Level of Analysis

When new technologies or systems are introduced the resulting improvements in productivity are likely to generate benefits for employers. Users who participate in the implementation of the new technology are likely to feel that they

Table 1. Equity-Implementation Model: Process Employed by a User in the Assessment of the Impact of an Implementation on His or Her Equity Status

Level of Analysis	Focus	Criterion	Operational Definition
1	Self	Change in equity status of self	Net change in equity status = Δ Outcomes − Δ Inputs
2	Self and the employer	Fair sharing of profits (benefits) between self and the employer	Δ Relative outcomes* of self vs. Δ Relative outcomes of the employer
3	Self and other users	Asymmetry in the impact on equity when compared with other users in the reference group	Δ Relative outcomes of self vs. Δ Relative outcomes of other users

**Change in relative outcomes may be expressed as:*

$$\Delta \text{ Relative outcomes on account of the change} = \frac{\Delta \text{ Outcomes}}{\text{Deservingness}}$$

and

Deservingness = Weighted average of outcomes expected based upon contributions (inputs), merit, equality, or other criteria.

should also share fairly in the benefits generated on account of users' and employer's collaborative effort in introducing the new system or technology. Users are likely to expect that the profits or benefits should be shared in proportion to the deservingness of each party.

Therefore, at the second level of analysis, a user is likely to compare the change in his or her relative outcomes with that of the employer. (Table 1 presents the conceptual formulation for an assessment of change in relative outcomes.) Leventhal[40] identified that in addition to the contributions criterion (i.e., inputs), individuals may also expect outcomes based upon criteria such as equality (a special case where inputs of all are considered the same), need, or merit (a relevant input). Therefore, the expected *deservingness* may be viewed as a weighted average of outcomes or gains expected based upon the criteria identified above. A user is likely to evaluate whether the gains have been shared between the employer and himself or herself in proportion to each one's respective deservingness. If the user feels that the employer (or the stock holders' group) has obtained greater relative gains compared to himself or herself or his or her group, the user is likely to become distressed and view the change as unfavorable.

In some instances the issue of procedural fairness may also be relevant. This can influence users' perceptions of fairness in the outcomes received due to the change.[34,41] For example, if some change is made on an arbitrary basis without users' involvement, users may perceive greater inequity compared to the introduction of the same change with proper bargaining or involvement process. Thus, the issue of procedures used in determining the relative allocation may also be relevant in some contexts.

Third Level of Analysis

At the third level of analysis, a user is likely to compare his or her relative outcomes with that of other users in the reference group. Some relevant questions would be: Does the new system impact all users similarly, or does the system result in increasing some users equity and lowering others? If a user feels that some other users or user groups have benefited from the new system while he or she has not benefited at all or not as much, the user is likely to experience inequity and assess the change as unfavorable. As discussed above, in some instances the issue of procedural fairness may also be relevant in determining the fairness perceptions in the relative allocation of resources to different user groups.

All three levels of analysis are likely to be important in determining the equity perceptions of a user. In addition, however, to considerations of his or her own outcomes and inputs, a user's equity perceptions may also be influenced by consideration of changes in the outcomes and inputs of his or her group or department. Strong group or departmental identification or affiliation may provide a frame of reference for the user to assess his or her own outcomes and inputs on the basis of the outcomes and inputs of the group.[34] The strength of group identification and its influence on equity perceptions may depend upon the nature of the group, the size of the group, and the type of outcomes and inputs being considered.

Users' assessment of changes in their own and others' inputs and outcomes is an important aspect of the determination of the impact of a change on equity. Different users are likely to consider different factors as relevant inputs and outcomes and may assign different values to different factors in arriving at the overall assessment of inputs and outcomes. Therefore, outcomes or inputs in the model represent benefits or contributions adjusted for their relative importance. For example, a benefit such as a $200 pay raise may be considered a greater outcome compared to a benefit such as a $200 increase in the budget. Even the same factor may be considered as an input by some users and an outcome by other users. For example, learning a new system (e.g., a popular word processing package) may be considered as an effort (i.e., an input) by an old employee who is unlikely to rise further or seek a change in his or her job. But learning the new system may be considered as the acquisition of a marketable skill (i.e., an outcome) by a young, ambitious employee who may be able to take advantage of the new skill in seeking a better job or promotion over time. Thus, it is not unusual that different users may evaluate the same change differently.

Table 2. Possible Changes in Outcomes and Inputs on Account of Implementation

Increase in Outcomes	Increase in Inputs
• More pleasant work environment	• More work in entering data
• Less tension, more job satisfaction	• More tension
• More opportunities for advancement	• Bringing higher level skills to the job
• Better service to customers	• Effort in learning a new system
• Recognition, better visibility	• Assignment of additional tasks
• Salary increase, grade increase, or higher-level title	• More effort in performing tasks in view of increased monitoring
• Increase in power and influence	• Need to spend more time
• Learning a marketable skill	• Fear of unknown (e.g., failure) and the resulting anxiety
• Reduced dependence on others	
• Usefulness of the system	
Decrease in Outcomes	**Decrease in Inputs**
• Reduced job satisfaction	• Ease of usage
• Reduced power	• Less effort
• Reduced bargaining power relative to the employer or others	• Reduced search for solutions or information
• Threat of loss of employment	• Reduced manual effort
• Loss of value of marketable skills	• Reduced cognitive effort
• Reduced importance, control	• Less rework due to fewer errors
• Increased monitoring	
• Reduced scope for advancement	
• More role conflict and ambiguity	
• Potentional failure in learning and adopting the new system	

The next section provides some insights into the nature of inputs and outcomes considered by users through three examples from different implementation contexts. The examples also help us understand the nature of the evaluation process and the three levels of analysis in determining the overall impact of change on equity.

Examples Illustrating the E-I Model

Example 1: Introduction of Computer Systems for Clinical Laboratory Personnel

In this section we examine the details of a case study reported by Kaplan and Duchon[42] involving computer systems implemented for reporting laboratory testing results in a hospital. Essentially, Kaplan and Duchon reported that in their questionnaire survey, differences in users' (clinical laboratory personnel) reactions to the introduction of new computer systems could not be explained by factors such as job characteristics, role conflict and ambiguity, departmental technology, and leader-member relationships. However, an analysis of data obtained from interviews and open-ended questions suggested that users' reactions to computer systems can be explained by their assessment of change in workload and change in results reporting.

Kaplan and Duchon used a process-oriented/product-oriented classification of users in order to explain their results. Users were viewed to be process-oriented if they believed that their workload increased and product-oriented if they believed that the service provided to doctors and nurses improved. However, the validity and generalizability of the new user characterization proposed by Kaplan and Duchon are questionable and have not been established.

Clinical laboratory personnel's different reactions to implementation can be explained by the E-I model without resorting to a new characterization of users. At the first level of analysis, all users are likely to view increased workload as an increase in inputs, though the magnitude assigned may differ. Users may also view as outcomes the reduction/elimination of interruptions on account of frequent telephone calls from doctors and nurses, better service provided, and faster delivery of results. Users are likely to vary in terms of value they assign to these outcomes. Some users may not even consider some of the above factors as relevant outcomes. Users who view increased workload as a significant additional input that outweighs the outcomes on account of reduced interruptions, better service, fewer errors, etc., are likely to perceive a loss of equity and view the new system unfavorably. This is supported in Kaplan and Duchon's study. Similarly, users who felt that outcomes on account of better service and less interruptions outweighed an increase in the inputs on account of increased workload should react favorably to the new system. This is also supported in Kaplan and Duchon's study. The magnitude of a user's positive or negative reaction to the system will depend upon the user's perception of the magnitude of net change in equity status. The E-I model also recognizes that users who feel that additional outcomes nearly match the additional inputs needed are likely to be relatively indifferent to the new system.

The model also suggests some other relevant inputs and outcomes not considered in Kaplan and Duchon's study.

These may be the effort to learn the new system, impact on wages or promotion prospects, computer monitoring of laboratory personnel, etc.

At the second level of analysis is the question of sharing the benefits of the new system between the hospital and the laboratory personnel. In this case it appears that the benefits may be somewhat ambiguous or intangible; therefore, the impact on equity at this level of analysis may not be very significant. It would not be surprising, however, if some clinical laboratory personnel expect higher pay or job levels due to their increased inputs and the benefits that the system may bring to the hospital.

The third level of analysis considers the issue of the impact of the new system on the relative equity position of different user groups. In this case, the new system appears to have nearly the same impact for all laboratory personnel. However, there are differences in change in equity position when doctors and nurses are considered. Some laboratory personnel are likely to view that the system increases their own workload while it benefits doctors and nurses substantially by improving communication and service to them. These lab users may not consider better service to be an outcome. This asymmetry in the benefits of the system is a possible source of distress for laboratory personnel.

While this example has been presented primarily as an illustration of the application of the E-I model, the model suggests an explanation of the findings of Kaplan and Duchon without relying on a new characterization of individuals. Unlike Kaplan and Duchon's analysis, the model also recognizes that some users may be indifferent to the new system. The model is also useful in identifying additional new factors that may be relevant in this context.

Example 2: Introduction of Computer Systems in Indian Banks

Employee unions in public sector banks in India have traditionally resisted the introduction of computer systems.[43] The E-I model can be used to explain the basis for this resistance. At the first level of analysis, bank employees may feel that the new system would require additional inputs in the form of learning and understanding new technology. Employees may also be apprehensive about possible negative outcomes such as computer monitoring and reporting of their activities, loss of promotion or advancement opportunities in a possibly thinned-out organization, and loss of control over work and bargaining power in an automated bank (i.e., reduction in impact of any likely strike action, as noticed in a recent strike by a telephone company's employees). Although, the government has given full assurances that none of the employees would be laid off or terminated, the possibility of transfer to an undesirable location also remains a likely negative outcome for some employees. Some likely increases in outcomes may be fewer errors, better customer service, and less fatigue. However, in view of stronger negative outcomes, the first level of analysis may return a possible net loss for employees.

At the second level of analysis, employees also realize that there is likely to be an increase in productivity and profits due to computerization. These benefits may be kept

Table 3. Suggestions for Improving Equity Impacts of Implementation

Strategy	Objective	Actions
Altering actual inputs and outcomes	Reduce users' inputs	• Well-designed training programs to reduce learning effort and frustrations
		• Help line/on-demand help
		• Extra temporary staff or overtime help during implementation
	Increase users' outcomes	• Positive equity through "royal/plush treatment" in training programs, design reviews, or briefing sessions
		• Praise, recognition, awards
		• Salary/grade increase
		• Job reclassification
		• Alleviate concerns about loss of employment, future prospects
Altering perceptions of inputs and outcomes (through suitable training, communication, and fair procedures)	Reduce users' inputs	• Emphasize learning new skills as outcomes rather than inputs
	Increase users' outcomes	• Emphasize the status and prestige of working in a modern environment with latest technology/innovation
		• Emphasize the outcomes on account of learning new marketable skills
	Employer's input and outcomes	• Explain employer's inputs and deservingness (e.g., risks, investment, R&D effort, etc.)
		• Explain the need to pass on the benefits to customers on account of the competition
	Other's inputs and outcomes	• Explain better-treated users' deservingness
	Users' perceptions of procedural fairness	• Fair procedures for user involvement, bargaining, and negotiations with user representatives

solely by the banks. Therefore, one of the conditions that employees have persistently advanced for accepting computer systems is that their salary scales and job classifications should be upgraded. This is to ensure that benefits of improvements in productivity are shared between employers and employees on what employees consider to be a fair basis. All of this appears to be in line with the second level proposed in the model. Further, if the changes are not negotiated with the unions through the normal bargaining processes, users may perceive procedural inequity and question the fairness of the outcomes.

At the third level of analysis, bank employees are likely to be concerned about the distribution of outcomes among different employees. Recently the Indian government agreed to pay increases for bank employees in branches where computers are to be installed. However, the bank unions, who view the deservingness of all bank employees to be the same, are insisting that the pay increase should apply uniformly to all employees, including those in the more than 95% of bank branches where computers are not yet scheduled to be installed.

Thus, the E-I model appears to explain the dynamics of employee resistance to computerization in Indian banks. All three levels of analysis appear to be relevant in this context.

Example 3: Introduction of Fourth-Generation Languages

MIS researchers have generally examined users' resistance to change. Reports of traditional programmers' resistance to the adoption of fourth-generation languages have also appeared in trade publications. The E-I model suggests the likely reasons. Fourth-generation languages may reduce the employment and advancement prospects for traditional programmers, resulting in negative outcomes. Traditional programmers may also fear the loss of market value and competitive advantage in a skill developed over time, which is also a negative outcome. Further, additional inputs may be required to learn fourth-generation languages. Some likely positive outcomes may be faster customer service and reduction in backlog and overtime. Because negative outcomes seem to dominate at the first level of analysis, there appears to be a net loss for traditional programmers. At the second level of analysis, the advantages of productivity improvement are likely to be mostly realized by employers due to the programmers' generally poor bargaining power. Finally, at the third level, when traditional programmers compare themselves with new programmers who learn fourth-generation languages, there is a stronger case for inequity. New programmers may be able to achieve the same outcomes as traditional programmers with far fewer inputs. Thus, many traditional programmers are likely to view the changes introduced by fourth-generation languages unfavorably.

Guidelines for Managing Change During Implementation

The E-I model suggests the importance of managing equity perceptions for successful implementation. A necessary first step may be to identify possible equity concerns of users with respect to the implementation. Actions to improve equity perceptions can be taken along two dimensions. Attempts can be made to improve equity by altering the actual outcomes and inputs of users, or by attempting to alter users' perceptions of their own and others' inputs and outcomes. Table 3 summarizes some of the possible actions that may be taken.

A number of steps can be taken to increase the *actual* outcomes of users. Users can be given additional outcomes in the form of an appropriate wage or job status increase. This may be particularly relevant if the new system involves development and use of higher-level skills by users. Users' outcomes can also be increased through changes in working conditions (e.g., flexible time schedule or reduction in the number of hours worked per week). Even if it is infeasible to provide the above outcomes, other actions can be undertaken. For example, the secretaries who learn a new word processing package can be given appreciation letters, special skill certificates, recognition, and small awards for cooperating in the implementation and for learning and using the new system. Users' fear of negative outcomes can be mitigated if employees are not discharged or laid off upon successful implementation. If assurances can be given to users as soon as possible on this issue, it may reduce their negative outcomes.

The strategy of positive equity (or over equity) can also be used to obtain user cooperation and a favorable response to implementation. Equity researchers have noted that positive equity is also discomforting to individuals.[33] Most individuals who experience positive equity in a relationship feel obliged to reciprocate by increasing the outcomes of the participants in the relationship.[44] The positive-equity strategy is used in many different contexts, ranging from advertising, sponsorships, donations, sales and marketing, to social contexts (compliments, entertaining, and gifts).

Some means for creating positive equity in the MIS context, such as giving appreciation letters, praise, recognition, and awards for help in implementation efforts, have been identified above. Additional steps may be to give users a posh treatment in training programs, design reviews, and briefing sessions. Many professional trainers would attest to the value of good food, plush surroundings, souvenirs, and elegant training materials in inducing positive equity among participants. Participants may view such training programs as an outcome rather than an effort or input.

Implementors can also take steps to minimize users' inputs. Some inputs that users have to provide in implementation include learning effort, additional tasks to be performed, and additional time requirements. The effort involved in learning new systems can be reduced through better teaching materials, personal attention, and on-demand help in diagnosing and solving problems. Implementors can also attempt to anticipate the additional workload and provide additional temporary help or monetary compensation (e.g., overtime pay). It may be easier to offer such help it it is already planned and budgeted for in the project cost.

The second strategy for improving equity is through altering users' *perceptions* about their own and others' inputs and outcomes. Equity researchers recognize the importance of training and communication in altering individuals' perceptions about inputs and outcomes. For example, Greenberg[33] discusses how supervisory training programs can lead the newly promoted supervisors to view their increased responsibility as a desirable goal, i.e., an outcome from the new job.

Training and communication are likely to be important tools for managing user perceptions about inputs and outputs and the distribution of benefits, particularly when users' perceptions may be formed in the absence of pertinent information. Users can be influenced to view learning as an outcome that will improve mobility and job prospects rather than as an input. The benefits of the new system can be emphasized in improving working conditions and quality of work. Training programs can also present the use of the latest technology and systems as outcomes for users.

The question of distribution of benefits among employer and employees is also important. Assuming that a fair attempt has been made to share the benefits, users can be convinced to view the company's survival and financial viability against the competition as a desirable outcome that would bring stability and security to their jobs. This should mitigate users' perception of unfair allocation. For example, no airline can survive without computerization of reservations. Therefore, in a highly competitive environment the employer may be forced to pass on the benefits of computerization to consumers. Properly disseminating this information should help mitigate perceptions of inequity by employees.

Users' perceptions of inequity can also be mitigated by explaining the deservingness of the better-treated user groups through suitable training and communication programs designed to reduce the perceived inequity. For example, in the clinical laboratory case study (discussed earlier), lab personnel may feel that the system favors doctors and nurses. Inequity on this account can be mitigated by explaining the deservingness of doctors and nurses by highlighting issues such as the very high cost of their time, time pressure, urgency of reports for saving patients' lives, importance of patient service for hospital business, etc. Thus, the E-I model suggests guidelines for focusing the objective and content of training and communication programs.

Finally, the establishment of fair procedures for determining the relative outcomes for different user groups and for the employer and users may also contribute to lowering the perception of inequity in the relative allocation of outcomes. Changes that are introduced through some fair procedure, such as bargaining with trade unions, negotiations, and user involvement and participation, are likely to be viewed better at least in terms of procedural fairness.

While the above analysis provides guidelines for managing implementation efforts, all implementation problems cannot be overcome. For example, the distress of inequity experienced by users who are about to lose their jobs due to the new system cannot be overcome despite the best efforts of implementors. The decision to terminate users may be beyond the control of implementors. However, to the extent possible, such extreme inequities should be avoided. Highly inequitable treatment of some users, such as termination, is likely to influence the equity perceptions of other users as well. Equity researchers have recognized the importance of group identification in determining an individual's equity perceptions.[34] When a user views other inequitably treated users as belonging to his or her group (e.g., clerical group, departmental group, lab technologists' group, etc.), even the better-treated users may develop perceptions of inequity. For example, if a clerical user is discharged, other clerical users in the reference group may also feel inequitably treated.

Conclusion

This report develops a theory-based understanding of information systems users' resistance to change. It describes a three-level process employed by users to evaluate a change in terms of its impact on their equity status. To assess the change in equity, users are viewed as evaluating their net gain based upon changes in their inputs and outcomes and comparing their relative outcomes with that of other users or user groups and the employer.

Contributions of the Research

The model provides an alternate theoretical basis for explaining and predicting resistance to change as well as different user assessments of the same change. The framework of analysis presented in the model is likely to be useful for practioners as well as researchers. Some guidelines for implementation identified in this report may be useful for managing IS implementation. Unlike some previous models, it can be applied to any level of change or implementation, ranging from the implementation of a word processing package involving one or more users to the implementation of large integrated systems involving many users in one or more departments.

Previous MIS research, based upon different theoretical perspectives, has made substantial progress in understanding users' resistance to changes introduced by IS implementation. Therefore, one important aspect of elaborating the contributions of this study would be to examine how this study improves upon previous work. The three levels of the E-I model should be appropriate for organizing this discussion.

At the first level of analysis, previous studies have identified and focused upon one or two specific factors at a time. For example, Davis[26] identified ease of use (less inputs) and usefulness (an outcome); Markus[19] identified power (an outcome); Turner[29] identified stress (a negative outcome) and job satisfaction (an outcome); Joshi[45] identified role conflict and role ambiguity (negative outcomes); and Kaplan and Duchon[42] identified improved customer service (an outcome) and increased workload (an input). This study suggests that these different factors should be viewed as inputs and outcomes and that users are likely to consider *all* of them together in assessing a change. The model identifies users' concern for an equity balance in their inputs and outcomes and the need for a net gain in equity balance for a change to be considered favorable. Models that consider only a few inputs or outcomes may be omitting important variables considered by users. The model clarifies the issue of relevant inputs and outcomes; therefore, the search for other relevant factors can be facilitated in future research. For example, it was noted that trained secretaries may view their higher-level skills as an additional input.

The second and third levels of analysis consider the interests of different groups. Previous MIS research has focused mainly on the third level of analysis. Markus[19] identified the issue of conflict and struggle among different user groups as providing motivation for resistance to change. Two key elements of this political perspective are: *conflict* between different user groups, and their attempts to gain *power* and other resources (outcomes). In the E-I model, conflict or struggle for power is not necessary. The E-I model describes a process of comparison that can occur with or without any direct interaction or conflict among different groups. Thus, the model makes less restrictive assumptions. According to equity researchers, the process of comparison is likely to be relevant in most settings, even among participants who do not interact directly. Therefore, the model has a wider applicability compared to the political perspective in explaining resistance to implementation. Further, the relevant issues for comparison should be the overall relative outcomes, which involve consideration of a wide range of inputs and outcomes, in addition to power and computing resources. The model also identifies the issue of comparison with the employer (which may include stock holders) in the second level of analysis. Comparison with the employer is not the same as comparison with managers or other users, as clarified in the example on Indian banks.

In the context of user acceptance of change, MIS researchers have identified the need for "an integrating paradigm to guide theory development and to provide a common frame of reference within which to integrate various research streams."[46] The E-I model has the potential to fulfill this need. It could provide an integrating framework for the implementation, user acceptance, and resistance to change research. While future research should assess this potential, there is some evidence to encourage consideration of the model for this role. The model provides an opportunity to integrate separate streams of implementation research, such as those pursued by Markus[19] and Davis,[26] under one framework. The relationship between the work of Markus and Davis has not been adequately recognized in the previous research, as evident from the fact that Davis[26] does not cite Markus.[19] The three levels of analysis proposed in the model consider a wide range of issues under a single framework and, therefore, offer additional insights into users' assessment of implementation, as suggested by the three examples discussed earlier.

Finally, the model also identifies the limitations of implementors in overcoming user resistance. It identifies the issues that may be beyond the authority of many implementors (e.g., the issue of pay raise or promotion). It may be necessary to involve the upper or top management to deal with some of these issues. An appreciation of user concerns as provided by the E-I model should be useful for management in arriving at better, well-informed decisions.

Limitations

The model needs to be tested in the MIS environment. The full range of inputs and outcomes considered by users also needs to be identified. Some limitations in the evaluation of inputs and outcomes should also be recognized. There may be some limitations and weaknesses in users' evaluation and decision-making processes that may limit the objectivity of users' assessment to some extent, although the extent of such difficulties remains to be fully understood. Possible fear and uncertainty about the nature of changes may also make it difficult for users to make an objective assessment. Users may also lack awareness of some outcomes and inputs. Therefore, as additional information becomes available, users may change their assessment. Users may also have mixed feelings about a given change. However, once a system is installed, it should be possible for users to make a better assessment. If the change is viewed unfavorably, resistance may develop and continue even after implementation.

Although the model identifies the relevant issues and problems considered by users, it does not mean that there is an easy solution to some of these issues. Issues such as fair sharing of benefits with employers and increasing the monetary outcomes of users may be beyond the control of many implementors.

Directions for Future Research

The E-I model presented in this report raises a number of research questions and opens up directions for future research. The model can be evaluated through case studies and empirical studies in field settings. Research may be conducted to identify the relevant inputs and outcomes considered by users in different implementation contexts. The relationship between perceived changes in inputs and outcomes and users' assessment of a specific implementation can also be investigated empirically. Development of suitable instruments is also needed to assess users' perceptions of equity impacts of an implementation.

The effectiveness of strategies for altering actual inputs and outcomes and strategies for changing users' perceptions about inputs and outcomes can also be investigated through suitable intervention in field settings and possibly through laboratory experiments.

References

A similar paper was presented at the TIMS/ORSA Joint National Meeting in Nashville, Tennessee, May 15, 1991.

[1]Clemons, E.K. "Information Systems for Sustainable Competitive Advantage," *Information and Management* (11:3), 1986, pp. 131-136.

[2]Joshi, K. "The Role of Systems Designers in Organizations," *Omega: The International Journal of Management Science* (18:5), 1990c, pp. 463-472.

[3]Bostrom, R.P. and Heinen, J. "MIS Problems and Failures: A Socio Technical Perspective: Part I: The Causes," *MIS Quarterly* (1:3), September 1977, pp. 17-32.

[4]Kumar, K. and Welke, R.J. "Implementation Failure and System Developer Values: Assumptions, Truisms and Empirical Evidence," *Proceedings of the Fourth International Conference on Information Systems*, Tuscon, AZ, December 1984, pp. 1-17.

[5]Lucas, H.C. "Empirical Evidence for a Descriptive Model of Implementation," *MIS Quarterly* (2:2), June 1978, pp. 27-41.

[6]Lucas, H.C. *Implementation: Key to Successful Information Systems*, Columbia University Press, New York, NY, 1981.

[7]Zmud, R.W. *Information Systems in Organizations*, Scott Foresman and Company, Glenview, IL, 1983.

[8]Ginzberg, M.J. "A Study of the Implementation Process," *TIMS Studies in the Management Sciences* (13), 1979, pp. 85-102.

[9]Ginzberg, M.J. "Early Diagnosis of MIS Implementation Failure: Promising Results and Unanswered Questions," *Management Science* (27:4), April 1981, pp. 459-478.

[10]Zmud, R.W. and Cox, J.F. "The Implementation Process: A Change Approach," *MIS Quarterly* (3:2), June 1979, pp. 35-43.

[11]Lewin, K. "Group Decision and Social Change," in *Readings in Social Psychology*, T.L. Newcomb and E.L. Hartley (eds.), Henry Holt and Company, New York, NY, 1952, pp. 459-473.

[12]Schein, E.H. "Management Development as a Process of Influence," *Industrial Management Review* (2:2), Spring 1961, pp. 59-77.

[13]Kolb, D.A. and Frohman, A.L. "An Organization Development Approach to Consulting," *Sloan Management Review* (12:1), Fall 1970, pp. 51-65.

[14]Wetherbe, J.C. and Dickson, G.W. *The Management of Information Systems*, McGraw-Hill, New York, NY, 1985.

[15]Bostrom, R.P. "A Socio-Technical Perspective on MIS Implementation," paper presented at ORSA/TIMS National Conference, Colorado Springs, CO, November 1980.

[16]Mumford, E. and Weir, M. *Computer Systems in Work Design: The Ethics Method*, Halstead (Wiley), New York, NY, 1979.

[17]Danziger, J.N., Dutton, W.H., Kling, R., and Kraemer, K.L. *Computers and Politics*, Columbia University Press, New York, NY, 1982.

[18]Kling, R. "Social Analysis of Computing: Theoretical Perspectives in Recent Empirical Research," *Computing Surveys* (12:1), March 1980, pp. 61-110.

[19]Markus, M.L. "Power, Politics and MIS Implementation," *Communications of the ACM* (26:6), June 1983, pp. 430-444.

[20]Pfeffer, J. *Power in Organizations*, Pitman Publications, Marshfield, MA, 1981.

[21]Keen, P.G.W. "Information Systems and Organizational Change," *Communications of the ACM* (24:1) January 1981, pp. 24-32.

[22]Kling, R. and Iacono, S. "The Control of Information Systems Developments After Implementation," *Communications of the ACM* (27:12), December 1984, pp. 1218-1226.

[23]Markus, M.L. "Implementation Politics: Top Management Support and User Involvement," *Systems, Objectives, Solutions* (1), 1981, pp. 203-215.

[24]Mumford, E. and Pettigrew, A. *Implementing Strategic Decisions*, Longman, London, England, 1975.

[25]Robey, D. and Markus, M.L. "Rituals in Information Systems Design," *MIS Quarterly* (8:1), March 1984, pp. 5-15.

[26]Davis, F.D. "Perceived Usefulness, Perceived Ease of Use, and Acceptance of Information Technology," *MIS Quarterly* (13:3), September 1989, pp. 319-340.

[27]Baroudi, J.J., Olson, M.H., and Ives, B. "An Empirical Study of the Impact of User Involvement on Systems Usage and Information Satisfaction," *Communications of the ACM* (29:3), March 1986, pp. 232-238.

[28]Nichols, M.L. "A Behavioral Analysis for Planning MIS Implementation," *MIS Quarterly* (5:1), March 1981, pp. 57-66.

[29]Turner, J.A. "Computer Mediated Work: The Interplay Between Technology and Structured Jobs," *Communications of the ACM* (27:12), December 1984, pp. 1210-1217.

[30]Ginzberg, M.J., Schultz, R., and Lucas, H.C. "A Structural Model of Implementation," in *Applications of Management Science: Management Science Implementation*, R. Schultz (ed.), JAI Press, Greenwich, CT, 1984.

[31]Swanson, E.B. *Information System Implementation: Bridging the Gap Between Design and Utilization*, Irwin, Homewood, IL, 1988.

[32]Walster, R., Walster, G., and Berschied, E. *Equity: Theory and Research*, Allyn and Bacon, Needham Heights, MA, 1978.

[33]Greenberg, J. "Approaching Equity and Avoiding Inequity in Groups and Organizations," in *Equity and Justice in Social Behavior*, J. Greenberg and R.L. Cohen (eds.), Academic Press, New York, NY, 1982, pp. 389-436.

[34]Joshi, K. "The Measurement of Fairness or Equity Perceptions of Management Information Systems Users," *MIS Quarterly* (13:3), September 1989b, pp. 343-358.

[35]Joshi, K. "An Investigation of Equity as a Determinant of User Information Satisfaction," *Decision Sciences* (21:4), Fall 1990b, pp. 786-807.

[36]Joshi, K. "A Causal Model of the Overall User Attitudes Toward the MIS Function: The Case of User Information Satisfaction," *Information and Management*, forthcoming, 1991.

[37]Joshi, K. "Understanding Information Systems User Behavior," working paper, School of Business Administration, University of Missouri, St. Louis, MO, 1990a.

[38]Adams, J.S. "Towards An Understanding of Inequity," *Journal of Abnormal and Normal Social Psychology* (67), 1963, pp. 422-436.

[39]Adams, J.S. "Inequity in Social Exchange," in *Advances in Experimental Psychology*, L. Berkowitz (ed.), Academic Press, New York, NY, 1965, pp. 267-299.

[40]Leventhal, G.S. "The Distribution of Rewards and Resources in Groups and Organizations," in *Advances in Experimental Social Psychology* (Vol. 9), L. Berkowitz, and E. Walster (eds.), Academic Press, New York, NY, 1976a, pp. 91-131.

[41]Leventhal, G.S. "Fairness in Social Relationships," in *Contemporary Topics in Social Psychology*, J.W. Thibaut, J.T. Spence, and R.C. Carson (eds.), General Learning Press, Morristown, NJ, 1976b, pp. 211-240.

[42]Kaplan, B. and Duchon, D. "Combining Qualitative and Quantitative Methods in Information Systems Research: A Case Study," *MIS Quarterly* (12:4), December 1988, pp. 571-586.

[43]Joshi, K. and Sauter, V.L. "Opportunities and Constraints in the Formulation of an Informatics Policy: The India Experience," *Information and Management*, forthcoming, 1991.

[44]Krebs, D. "Prosocial Behavior, Equity, and Justice," in *Equity and Justice in Social Behavior*, J. Greenberg and R.L. Cohen (eds.), Academic Press, New York, NY, 1982, pp. 261-308.

[45]Joshi, K. "Role Conflict and Role Ambiguity in Systems Design," *Omega: The International Journal of Management Science* (17:4), July 1989a, pp. 369-380.

[46]Davis, F.D., Bagozzi, R.P., and Warshaw, P.R. "User Acceptance of Computer Technology: A Comparison of Two Theoretical Models," *Management Science* (35:8), August 1989, pp. 982-1003. ■

LAN Implementations: Case Studies

In this report:

Datapro Summary

The case studies presented here demonstrate many of the potential benefits inherent in LAN technology. All three of the profiled companies—a natural gas products transporter, a medical clinic, and a gas marketing company—implemented LAN systems to conduct mission-critical operations (those crucial to the company's business). All three obtained better productivity with their new LAN systems compared to their previous systems, contributing to the continuing success and growth of their business. In each case, a LAN has provided benefits that other solutions would not have been able to provide for the same price.

Company A: Gas Products Transporting Company

Company Background

Company A, the largest natural gas products transporter in the United States, buys and sells a variety of energy products with customers and suppliers around the world. They have major corporate offices in Houston, TX; Paris, France; and the Netherlands.

Business Need

Company A needed to integrate their international liquid petroleum gas (LPG) trading business with their ship management business. It required an integrated, automated group of information systems to track its international transactions, including all trading deals, contracts, prices, suppliers, receivers, ports, and vessels. In addition, it needed a ship management system to store information not only for vessels owned by the company, but for all vessels that could potentially carry LPG products.

—By Ben Mayberry and Elisa Garza
Business Systems Group, Inc. (BSG)

Adding to the challenge, it needed the integrated system up and running almost immediately to coincide with the start of peak demand periods for LPG (the beginning of winter in the Northern Hemisphere).

Company A had already implemented a number of systems on a local area network, but these systems were not integrated and each required expansion. Integration would help them run efficiently (perform their duties easier and faster than before) by allowing all employees to access a single, complete database of information in a graphical format. It also needed to implement the ship management system. When the company began working with a consulting firm toward a solution for this set of problems, it realized that a LAN-based solution was not only the answer to its computer system needs, but also the best solution from a budget perspective and one that would allow easy adaptability for change and growth.

The integrated system needed to perform four groups of functions:

- Contract Profile
- Supply/Demand
- Cargo Recap
- Ship Management

The individual duties in these function groups included:

- contract administration, including record-keeping and the sending of contracts over telex
- voyage calculation, to determine the best economic alternative for shipping
- ship chartering
- product accounting
- deal economics, to determine the productivity of their transactions
- customer invoicing
- invoice payment
- storage management

The first three function groups (Contract Profile, Supply/ Demand, and Cargo Recap) existed in a LAN environment, but each system stood alone. The fourth function, Ship Management, did not yet have mechanized support. A new system was needed to integrate and enhance these organizational systems, allowing for the overlapping and sharing of data, and the capability to add new systems as the need arose.

System Design

Company A worked with the consultants they hired to develop a customized, integrated system that operated from a central database server. This system was to operate on the pre-existing LAN. The consulting team used an iterative prototyping approach to develop the system. They divided the project into four phases:

- system design
- prototype development
- system implementation
- system support

During the first phase, system design, the consulting team evaluated the existing systems, consulted with the key users to determine functional requirements, designed the database structure, and defined the hardware and software requirements. Two months after the start of the project, the consulting team had developed a prototype system and requested that key users review and test the system. The team also revised the system based on user feedback and suggestions. Less than four months after they began work on the project, the consulting team implemented the system, meeting the requirement that the system be in operation for the peak demand period. By this time, they had completed development and user testing of the system, developed and conducted training programs for all of the affected users, and converted all information from the existing systems to the new database. During the first month of system operation, the consulting team provided system support, including technical, training, and general user assistance.

The existing technical environment was enhanced to use the Novell NetWare 386 network operating system with two Compaq SystemPros, one as a file server and another as the database server running Oracle Server for NetWare. The PCs used with the previous systems were not changed.

Results

Company A has gained many advantages from the changes to an integrated LAN solution. In the short term, this solution is much more cost effective than a mainframe solution, both initially and for ongoing costs. In addition, quick system development and early training classes saved costs for consulting and employee learning time, allowing for significant productivity improvement in the short term.

But the real benefit is long term. This gas products transporting company now has an integrated trading, ship management, operations, and product accounting system that allows everyone to look at the same data, produces accurate and timely invoices, generates timely management reports, and allows employees to perform their jobs quickly and effectively. In addition, this system can be easily expanded as new needs arise. Because the system is constructed from industry-standard tools, leading-edge and developing technologies and extensions to the current system can be implemented in a cost-efficient and timely manner.

For instance, the following functions can be easily added to the current system:

- additional reporting
- crewing and ship's stores support
- maintenance and repair support
- capacity to store and display images of key ship components and systems.

In addition, the system will also be compatible with developing technologies such as image processing, electronic data interchange (EDI) through wide-area communications, and real-time data acquisition and graphical presentation through graphical user interfaces (GUIs).

Company B: Medical Clinic

Company Background

Company B, a medical clinic, was founded by a doctor as a specialty practice in infectious diseases and internal medicine, focusing specifically on several areas, including immune deficiency states and travel medicine.

Business Need

Seven years after its founding, the clinic decided to expand its patient care operations and develop a new dimension to its services that would address the needs of businesses whose employees frequently travel internationally. In particular, it planned to expand from a single-site into a multisite company consisting of several legal entities, including a patient care facility and a pharmacy. Company B did not feel that the minicomputer system it had in operation met its needs prior to the expansion, nor would it do so after the expansion because it was difficult to use and could not be enhanced to accommodate new functions.

Company B needed an integrated system to support three main areas.

Clinical care to support the delivery of efficient and effective clinical care.

Business office and administrative support which was responsible for rapid collection of payment and efficient operations.

Marketing to increase profitable market share and promote the business.

Some of the specific functions they required of the system were:

- order entry of patient registration information and history
- dictation support
- scheduling of patients, physicians, and nurses
- printing of prescriptions, hospital admission instructions, test vial labels, and patient instructions
- treatment planning and development of treatment protocol
- billing
- accounts receivable management
- test scheduling, monitoring, and control
- electronic mail
- inventory management
- performance monitoring
- marketing and accounting analysis
- external hookups to medical software services
- monitoring of client, referring doctor, and patient relations
- referral letter, client-specific document, thank you letter, and follow-up notice generation
- profit analysis by category, such as referral source

Company B wanted an integrated, user-friendly system that would support its immediate and future growth plans. In addition, it desired a system that used its existing supply of PCs, would require little formal training of employees, provided remote access, and had structured security features. It requested that the new software minimize typing (by using menus), be adaptable to changing business functions, and allow for input in foreign languages.

System Design

Based on a long list of needs and requirements, the consulting team the clinic hired recommended the use of GUI technology running on a PC LAN. At the time, no medical office management applications existed that used GUI technology. After discussing the alternatives with the consulting team, Company B agreed that a custom-designed system would best meet its needs.

The consulting team then divided the project into four stages, to be completed separately:

- initial phase, encompassing two small immediate projects
- high-priority phase
- patient-care phase
- accounting phase

In the initial phase, the consulting team installed the dictation equipment and designed the procedures for dictating the information obtained by the doctors during their hospital practice.

The high-priority phase was completed six months later. It included the patient registration and scheduling systems, on which all the other systems were dependent, and all functions previously performed by the old system. Because all of these systems were implemented together, employees did not have to run old and new systems simultaneously, and they were able to improve productivity before the remaining systems were implemented.

Three months later, the patient-care phase was completed. New functions added in this phase included order entry, laboratory results processing, additional dictation support for the billing system, and hospital admission procedures.

The final phase was completed three months later, a year after the initial project began. The system now supported billing; accounts receivable; independent pharmacy operations; cost, profitability, and market analysis; and accounts payable and general accounting packages. Leaving these functions for the final phase allowed the consulting team to implement some GUI accounting applications that were not available when the project began.

The consulting team made use of the medical clinic's existing supply of PCs by installing math coprocessors to improve the speed of calculations and graphical displays. These computers were connected as a LAN using an unshielded twisted-pair (UTP) Ethernet topology. The LAN is managed by Novell's NetWare 386 network operating system. To provide for remote access, a communications server with high-speed modems and a network interface card (NIC) runs Novell's NetWare Asynchronous Communications Server (NACS) software. The custom applications were programmed in Microsoft Windows using the C programming language. Information is stored in a database from Gupta Technologies called SQLBase. Employees also use Gupta's SQLR to create custom reports.

Results

Early implementation of the high priority systems allowed Company B's employees to begin increasing productivity six months before the project was completed. This was achieved because the new systems were easier to use than the old one. In addition, the clinic has only begun to realize the benefits they have gained from accurate revenue and profitability reporting. They can calculate the profitability of each segment of their business, each referral source, each physician, and each nurse. Also, their overall system was designed with growth in mind, using leading-edge technology and products. As the clinic's business expands, their LAN is ready to expand and grow with them.

Company C: Gas Marketing Company

Company Background

Company C is the largest independent gas marketing company in operation today. Almost four years ago, it employed only a dozen people using six PCs. At that time, Company C hired a consulting company to develop an integrated computer system that would grow with it. The consulting team set up a PC LAN with custom applications to meet the company's needs. Today, Company C employs 300 people in 12 district offices around the country, and still uses an extended version of the initial LAN, with the same applications.

Business Need

Company C needed an integrated system for tracking the buying, transporting, and selling of gas from bid origination through final disposition and accounting. It communicated a set of initial needs to the consulting team it hired. The solution needed to support critical business functions such as the tracking of a prospect through final account balancing, and providing market data. Company C also wanted something that was powerful and easy to use.

The consulting team divided the critical functions into three main areas: marketing and supply, transportation and exchange, and accounting.

Some of the procedures required of the system were:

- market- and supply-opportunities tracking
- marketing accepted deals
- storing historical information
- market and supply analysis
- nomination setup (choosing a pipeline to move the gas)
- tracking the daily flow of gas
- calculation of transportation and fuel costs
- end-of-month volume and sales reports
- tracking pipeline imbalances

System Design

The consulting team chose a PC LAN for three reasons: the LAN would support the data the company had acquired over the primary phase of its business; potential for LAN expansion was limitless; and it was the most cost-effective solution.

This report was prepared exclusively for Datapro by Ben Mayberry and Elisa Garza of Business Systems Group, Inc. (BSG). Ben Mayberry is BSG's director of consulting. Elisa Garza is a staff consultant and technical editor for BSG Communications. BSG, based in Houston, TX, is a national systems integration company specializing in business solutions based on client/server, network computing technology. BSG can be reached at (713) 965-9000.

Company C began operation three years before, when deregulation in the gas pipeline industry opened up commercial opportunities. Because gas marketing was a relatively new industry, no off-the-shelf software packages were available to meet its needs. The consulting team chose to custom design the required applications in the Advanced Revelation DBMS (database management system).

The entire project was completed in eighteen months. By then, Company C had a total of 85 employees in four regional offices. The consulting team set the company up with Compaq desktops and portables as nodes on the LAN and a Compaq SystemPro fileserver with a 130MB disk drive. The LAN runs with Novell's NetWare network operating system. The remote offices are connected to the Houston office through a WAN (wide area network).

Results

In the gas marketing business, all companies provide the same product. The only difference among the various companies is the quality of service that they provide. Company C's integrated system enables it to provide accurate, timely service on a consistent basis. Its reputation for such service has contributed to the company's growth. At the time that it implemented the integrated system, Company C estimated its growth to be 15% annually. Its actual annual growth rate since then has been almost 50%. Company C's business has grown very quickly; however, its staffing requirements have not. The efficiency and speed of its computer system has allowed the company to grow without significantly increasing the number of employees.

Summary

As is evident from these three case studies, all of these companies have benefited from implementing LANs. In each case, the company is prepared for future growth and the implementation of new technology. These companies are not exceptions in the business world. LANs have become a vital part of the business structure in almost every industry. ■

Project Management Software

In this report:

Datapro Summary

Although by some standards project management software is still considered a specialty category, PC-based systems have found a home in mainstream business software libraries everywhere. Keeping track of every aspect of complex projects, from tasks and resources to costs, can be extremely confusing. With roots in the mainframe world, project management software designed for the PC lowers the entry costs of computer-aided project planning and generally offers a simplistic user interface. Project management software is perceived as more difficult to use than most microcomputer applications, because its techniques and methodologies are not always familiar to project planners. Although project management software can help to plan and control a project, it is only a tool. The actual success of a project's completion is ultimately the project manager's responsibility.

Technology Basics

A project is typically coordinated by a project manager, who is responsible for ensuring that tasks are successfully completed on time and within a specific budget. Keeping tabs on every aspect of a project, from tasks and resources to costs, can be a daunting challenge for even the most seasoned project planner. That is why PC-based project management can be such a valuable productivity tool. The software displays graphical presentations of a schedule so that components can be readily understood, calculates the critical path, notifies the user of resource conflicts and time constraints, and tracks costs.

Is project management software for everyone? For relatively small projects, the software may not be needed. But for projects that involve more tasks, people, equipment, and deadlines than can be comfortably handled with a calendar and a pen, project management software can surely help.

Planning a project on a PC offers managers the ability to accomplish quickly and easily what might ordinarily take days, or even weeks, to execute manually. The software allows the planner to create "what-if" scenarios, manipulating planned time and dollars-versus-actual time and dollars. Controlling the flow of a project is the key to its ultimate success.

Computerized Project Management

The term "project management" was coined in the 1950s, and at that time its practice was generally reserved for manually scheduling and tracking large-scale government projects. The first project management discipline—Program Evaluation and Review Technique (PERT)—was introduced in the late 1950s when the U.S. Navy developed its Polaris Missile. This methodology was designed to aid the Navy in analyzing critical relationships between tasks. It consisted of assigning each task three time estimates—a worst case, a best case, and a most probable—for completion. PERT then used statistical probabilities to calculate expected task durations. Over time, PERT charts evolved into network diagrams that illustrate the relationships and dependencies between tasks.

Two other key techniques were used in early project management efforts. Critical Path Method (CPM) was used by DuPont Corporation in the late 1950s for scheduling its construction projects. The Gantt chart was created by Henry Gantt, a disciple of the scientific school of management. CPM calculates the duration of individual tasks and their interdependencies with an eye on the scheduled finish date, while the Gantt charting method illustrates tasks across a time scale.

For decades, project management was exclusively a mainframe-based operation, primarily used for extensive defense and civil engineering projects. The software was able to rapidly recalculate schedules and networks with each alteration in the plan. It also allowed managers to experiment with the schedule and create "what-if" scenarios. Unlike most of today's project management users, the professionals who operated the mainframe-based systems were called project management specialists.

PC Project Management

Harvard Project Manager, introduced in 1983, was the first PC-based project management program. This product paved the way for the first generation of project management programs. With the inception of these programs, complex techniques were taken from specialists and placed into the hands of anyone with access to a personal computer. For the most part, however, these early products lacked sophistication and "real-world" adaptation capabilities.

A second wave of project management programs was introduced in the mid-1980s. As microcomputer capabilities expanded, so did the power of project management programs. State-of-the-art interfaces, more project adaptability, and increased technical boundaries were offered. Project management vendors, entrenched in a features war, tried to outpace one another with new capabilities.

We are now witnessing a third generation of project management programs. There are some basic differences in features and approach between products, but the race to introduce new capabilities has slowed down. The emphasis seems to have shifted from technical specifications to user interfaces and ease of use. Even programs that are considered low end and relatively simple to use are able to handle more complex projects with each update. Microsoft Project, Symantec Time Line, and Computer Associates SuperProject Expert are examples of low-end products that offer power formerly found in mainframe computer descendants. Moreover, these programs offer the typical PC interfaces and prices that make them attractive for management use.

Project Management Concepts

A typical project comprises time constraints, budgets, estimates, and charts. Successful project planning consists of the following: anticipating the steps required to complete a project (including specific tasks and sequences); developing a schedule for these tasks and specifying deadlines; assigning resources (people and equipment); and estimating costs.

Tasks

Project management vendors refer to a project as any job that can be segmented into a series of smaller activities, or "tasks." Herein lies the basic purpose of project management software: to schedule and track each task as it occurs to ensure that the job flows smoothly and within budget. A less obvious, but very important, benefit of this software is that it forces the manager to sit down and map out a project's schedule, giving careful thought to the sequence of events and the estimated time that each task should consume. In this way, project management software serves as a thought processor.

What-If Analysis

One benefit of this software is that if the manager specifies a task's duration, the program will automatically calculate the start and finish dates. The software also usually gives the earliest and latest possible start dates for each task, calculated to ensure that the project is completed on time. Depending on the complexity of a project, if done manually, these calculations could take hours, sometimes days, to complete. The software is also designed to let the manager create "what-if" scenarios. For example, it can help predict the impact if, for instance, Task G is completed 10 days later than expected. How will other tasks be affected? A project management package should present answers to such questions.

Benefits

With its emphasis on time and how each task affects the final deadline, CPM is the leading method of project planning and scheduling. If Task G is delayed, it jeopardizes the completion of the entire project. The critical path identifies and shows what effect such a delay would have on the deadline date.

Most project management programs offer a variety of charts to help visually depict a project from different vantage points. Gantt, PERT, and resource histogram charts are most often used in project management. A Gantt chart shows tasks as bars on a horizontal time grid, while a PERT chart displays the relationships and interdependencies between task by means of connected boxes. Both chart types have their benefits. Gantt charts are useful for showing the duration of tasks but do not show task relationships. PERT charts, or network diagrams, are most useful for showing both durations and task relationships, but do not illustrate timing relationships. Subsequently, managers like to toggle between charts to get a bird's-eye view of an entire project.

The people or equipment that act on a task are "resources." Many project management programs provide "resource leveling" capabilities to alert a manager of resource conflicts or overallocation of resources. "Resource-limiting" leveling delays tasks pending the availability of the required resources. "Time-limited" leveling only delays tasks if they have "float," extra time before the task's delay affects another task.

Selecting a Project Management Program

Although all businesses and situations are different and the needs of users vary tremendously, there are several basic steps and issues to consider when selecting a project management program. The most efficient way to sort through the market is to classify programs by price.

High-end programs, or those products which generally cost more than $1,000, most closely parallel mainframe software programs in functionality and adaptability. These programs offer more in the way of technical boundaries (e.g., number of tasks, resources, etc.), budgeting, cost analysis, and reporting capabilities. High-end programs are suitable for experienced project managers with large-scale, complex projects to handle.

Some of the most popular project management programs are the low-end systems. These programs cost less

Project Management Software Glossary

Most project management software programs use the same terms. This glossary explains some of the terms associated with microcomputer project management packages.

activity—any work or single job that must be done in order to complete a project; synonymous with task.

activity coding—attaching identifiers to tasks to simplify sorting or selecting specific information; often used in conjunction with Work Breakdown Structures.

allocation—the act of designating resources to tasks, or costs to resources/tasks.

arrow diagram—a graphical method of depicting precedence between the tasks in a project; tasks are represented as arrows, and circles indicate milestones.

baseline—the original schedule as planned; comparing the actual versus the baseline schedule is a good way to measure a project's flow.

calendar—a feature that enables users to modify and customize a working schedule on a monthly basis to show working and nonworking days.

Critical Path Method (CPM)—a project management principle that leads the user through the scheduling process with the emphasis on time and how each task corresponds with the deadline. The "critical path" is the task sequence that consumes the most time; if a task on this path is delayed, it jeopardizes the completion of the entire project.

critical task—any task whose delay would cause the entire project to miss its deadline.

deadline—refers to the final date by which a project is scheduled to be completed.

dependency—the logical relationship between tasks; a dependent task is one that cannot be started until its predecessor has been completed.

duration—the amount of time required to complete a task; in most programs, task duration is expressed in terms of hours, days, or weeks.

earned value analysis—a computation of the cost of work satisfactorily completed; the basis of techniques developed by the Department of Defense.

elapsed time—the duration of a task which includes both working and nonworking days.

filter/sort—a feature that enables the user to sort project-related data in alphabetical, numerical, chronological, or other order.

float/slack—the amount of time a task can be delayed before an entire project is delayed.

Gantt chart—a graphical representation of a schedule showing tasks as bars on a horizontal time grid.

histogram—a graphical depiction of a resource's work load.

milestone—a starting or stopping point for a group of tasks within a project, representing a checkoff point or short-term goal.

network diagram—a chart that displays all the project tasks and how they are related to one another; a PERT chart.

parallel tasks—two or more tasks that are scheduled to occur at the same time, although they do not necessarily begin and end at the same time.

partial dependency—the specified lead or lag time between two tasks.

predecessor—a relationship whereby a task must be started or completed before another can be started or completed.

Program Evaluation and Review Technique (PERT)—one of the two most popular project management principles; a PERT chart graphically illustrates the relationships and dependencies between tasks by means of boxes and adjoining lines.

report generation—the ability to create a variety of project-related hard copy reports (e.g., cost, actual-versus-planned, task detail).

resource—anything that works on a task; usually a person, a group of persons, materials, or equipment.

resource leveling—a feature that alerts the user to the overallocation of resources and then redistributes the work load to correct the problem.

schedule—the design of a project's course in terms of a proposed sequence of events.

sequential tasks—tasks which follow one another in a project schedule.

successor—a task whose start time depends on the completion of one or more predecessors.

task—an activity (a job or work) that must be done to complete a project. A project comprises a series of tasks.

Work Breakdown Structure (WBS)—a hierarchical method of scheduling that divides a project into sections consisting of related tasks; usually constructed in either an outline form or as a flow diagram.

workday—a day that the program considers when calculating a project's schedule; it will ignore weekends, holidays, and other user-specified nonworkdays.

---

than $1,000 and, although they do not try to emulate mainframe programs, offer enough features to satisfy a broad range of managers. Low-end programs are typically easy to use and are geared to the general business user who has little formal project management software experience.

After deciding on a price range, selecting the ideal program becomes more difficult. This market, with no real industry standard, is expanding at a respectable pace, making nearly every project management program a candidate for a prospective buyer. Keep the following guidelines in mind to narrow the field.

The User: If the person is not familiar with project management or personal computers, then select a product with a reputation for ease of use. Fortunately, many of the low-end systems offer a winning combination of ease of use and solid performance. If the individual is a project management specialist or will be involved in planning complex projects, a high-end product would be more appropriate.

Number of Tasks and Resources per Project: If the project will span, for example, two years, the chances are that a high-end product with expansive technical limitations will be needed. A manager should not have to worry about the number of tasks or resources a product can accommodate when developing a complex project.

Reporting: Most project management programs offer a variety of printed reports—task detail, resource detail, cost tracking. If presentations must be made to management, investigate a product's capability to generate charts, as well as its support for laser printers, plotters, and even film recorders. If project cost is a priority, make sure the program can generate a range of cost-related reports. Aside from fixed report formats, most programs offer the capability to create customized reports.

Cost Allocation and Budgeting: If cost analysis is a priority, the program should supply planned budget versus actual costs, prorating, both fixed and variable costs, and earned value analysis.

File Compatibility: With project management software, it is important to be able to transfer project/cost information to a spreadsheet program to design detailed financial models. Also, can charts be exported to a word processing program? Does the product offer a link to a mainframe or high-end project management program?

Lastly, as with any software, try to get a hands-on demonstration. Create a simple project schedule that simulates an actual situation as closely as possible. There is no better way to find out whether a particular product is satisfactory. ■

LAN Design Concepts: Ethernet, Token Bus, and Token-Ring

In this report:

Datapro Summary

The first nonproprietary local area network, Ethernet revolutionized the LAN industry. In the early 1980s, Digital Equipment Corp., Intel, and Xerox issued a set of specifications for Ethernet that still remains a LAN standard. IBM's Token-Ring Network is direct competitor of Ethernet. The evolution of Ethernet and Token Ring is explored along with technical explanations of their basic components and associated LAN standards.

Ethernet

May 22, 1973, does not quite have the ring of July 4, 1776, but it is a significant day in network history. On that date, Robert Metcalfe wrote a memo at Xerox's Palo Alto Research Center and used the word *Ethernet* while explaining the principles behind this new type of local area network. That same year Xerox began producing Ethernet network interface cards for its Xerox Alto PC.

Intel provided the chips necessary for the network hardware, Xerox provided the software, and Digital Equipment Corporation was prepared to run this new network on its minicomputers. In September 1980, these three companies released a set of specifications for Ethernet that is now referred to as "Ethernet version 1." A second version of Ethernet ("Ethernet version 2") was released November 1982. Because both versions still are found in the field, it is important to know that significant differences exist between them. The line-idle state was changed from 0.7 volts idle in version 1 to zero volts idle in version 2. The interface coupling specifications of these two versions also differ. A controller designed for one version will not work with a transceiver designed for the other version.

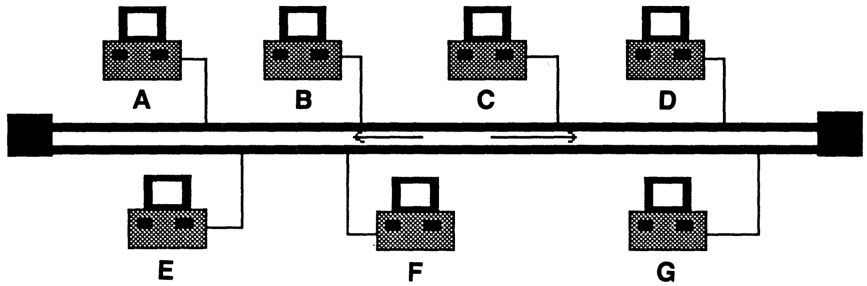
Traditional Ethernet is associated with the version 1 specifications, which defined the network's physical medium (thick coaxial cabling), the network access control method (carrier-sense, multiple-access with collision detection (CSMA/CD), and the speed (10 megabits/second transmission). The set of specifications also described the size and contents of an Ethernet packet (72 bytes to 1526 bytes) as well as the method of encoding data (Manchester encoding). Shortly after this release of Ethernet specifications, the IEEE 802 committee began deliberations on the development of a set of international, non-proprietary standards for LANs. Given the industry prominence of the three major founders of Ethernet, it should come as no surprise that one standard, IEEE 802.3, is so close to Ethernet (version 2) that it often is described as the "Ethernet standard" even though there are differences that are discussed in this report.

Carrier Sense Multiple Access With Collision Detection (CSMA/CD)

The IEEE 802.3 committee developed a protocol virtually identical to Ethernet's for describing how multiple workstations can access a network when they need to transmit information. *Carrier Sense Multiple Access with Collision Detection* (CSMA/CD) dictates that a workstation wanting to use a

This Datapro report is a reprint of book # 3577 "Ethernet and Token Bus", pp. 89-107; and "Token Ring Network and FDDI", pp. 109-131, from *Linking LANs: a Micro Manager's Guide* by Stan Schatto. Copyright © 1991 by Windcrest, an imprint of TAB Books, a division of McGraw-Hill, Blue Ridge Summit, PA 17294 (1-800-233-1128 or 1-717-794-2191). Reprinted with permission.

Figure 1.
An Ethernet or IEEE 802.3 Network



Workstation "F" sends a message to workstation A.
Notice that the message is transmitted in both directions.

network must first listen to the network to see if it is busy. If it does not detect any signals, it begins transmitting its message. This workstation continues to listen for possible network collisions while sending its message.

If a collision is detected, the sending workstation backs off and transmits a special *jam* signal to let network users know that a collision has taken place. The receiving station normally discards the contents of the partial message it has received, and all network workstations wait a certain randomly selected amount of time before any station begins transmitting again. Each network interface card has a different amount of time programmed for waiting. This time increases if a collision occurs the very next time the same message is transmitted.

Collisions are inevitable with an Ethernet or 802.3 network because of the very nature of CSMA/CD. There is a time interval between the time a workstation listens to see if any other workstation is using the network and the time it actually begins transmitting. It is entirely possible that a workstation further down the network has begun sending but that the signal has not yet reached this workstation. Because the IEEE 802.3 standard describes the same bus network topology used by Ethernet, on both networks each workstation broadcasts its message in both directions on the network. Figure 1 illustrates this method of transmission.

This Ethernet type standard describes a *contention* network, a type of network in which more than one workstation must contend or compete for use of the network. While collisions are inevitable given the network's architecture, the designers assume that the 10 megabit/second transmission speed ensures that even with repeated collisions, users will not experience any noticeable delay. When the number of collisions does cause noticeable delays in response time, it becomes a network management issue.

The 802.3 and Ethernet Frame Format Differences

Figure 2 displays the 802.3 frame format. The Preamble consists of 56 bits of alternating 1s and 0s which provides a method for two networks to synchronize. The Start Frame Delimiter (10101011) signifies the beginning of a frame of information. The Destination and Source fields come from the LLC frame. The Length field indicates the length of the information field that follows it. Data can range from 46 to 1500 octets in length. The Information field is followed by a Pad field that ensures that the frame meets the minimum 802.3 length standards. Finally, a Frame Check Sequence field provides information for error checking.

The major difference between the 802.3 frame and the traditional Ethernet frame is that Ethernet does not have a two-byte length field because the length is fixed. Instead, it has a two-byte Protocol Type field that indicates which higher-layer protocol is used in the data field. You are likely to see some familiar names in this field. A value of hex 0800, for example, indicates TCP/IP, while a hex 0600 indicates XNS. Matching Ethernet and 802.3 transceivers (devices that actually transmit data from the network interface cards to the media) will result in network errors since an 802.3 or Ethernet node will misinterpret a message intended for the other type of device. Pin layouts are different for Ethernet and 802.3 transceivers. Ignoring this distinction will often result in 802.3 nodes becoming overloaded handling broadcast Ethernet messages and users' screens freezing. Guess who the end users are going to ask for help?

802.3 as an Evolving Standard

While standard Ethernet specifies only 50-ohm coaxial cabling, the IEEE 802.3 standard currently supports several different types of transmission including baseband and broadband coaxial cable and twisted-pair wire. The cabling chosen varies according to the recommended maximum distance. A version known as StarLAN offered by a number of vendors including AT&T provides one megabit/second transmission speed over 500 meters (1Base5). Thick coaxial cable (50-ohm) has a 500 meter limit, so we refer to this 802.3 standard as 10Base5, meaning 10 megabits/second baseband coaxial cabling with a 500 meter limit. Thin coaxial cable can carry a signal 185 meters (*10Base2* or *Cheapernet*) while unshielded twisted pair (10BaseT) is recommended up to 100 meters.

The older StarLAN 802.3 specification for a 1 megabit/second network with a 500 meter maximum length is known as 1Base5. Because the IEEE 802 subcommittees continue to meet as new technology develops, the standards continue to evolve. Since the 802 standards utilize a layered set of protocols very similar to the OSI model, it is possible to add to the MAC layer without having to change anything in the LLC layer.

Figure 2.
The 802.3 Frame Format

Preamble	Destination address	Source address	Length	Data	Frame check sequence

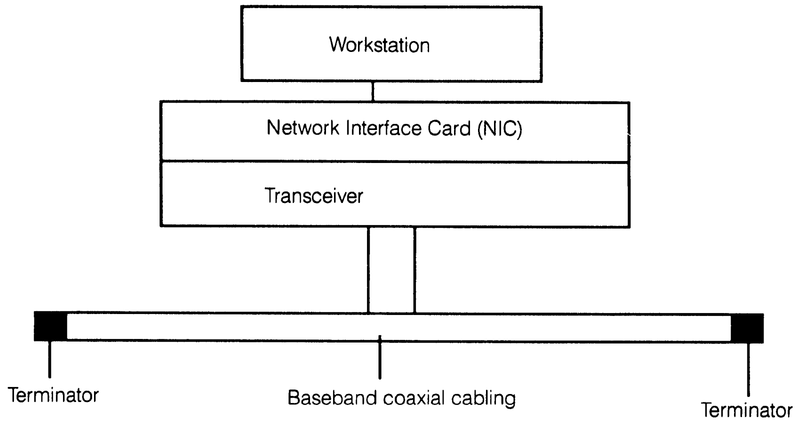


Figure 3.
Ethernet Architecture

A Closer Look at an Ethernet Network

Before you look at some variations, examine the components comprising the traditional Ethernet or 802.3 LAN; just as the Colt 45 became known as the gun that won the West, Ethernet is the LAN that made local area networks respectable to corporate managers. They could install it and depend on it without worrying about being fired for buying fringe technology.

An Ethernet workstation contains an Ethernet-specific network interface card (NIC) which has its own intelligence. This NIC is responsible for handling collision management and encoding and decoding signals. With thin coaxial cabling a transceiver is part of this NIC as illustrated in Figure 3, while transceivers are external on thick coax networks. The terminators complete the segment's electrical circuit.

A single segment can hold a maximum of 100 workstations, and multiple segments can be linked by repeaters. Up to 1024 workstations can exist on a single Ethernet network with a maximum of two repeaters in the path between any two stations.

The transceiver actually generates the electrical signals to the coaxial cabling and maintains their quality. Conversely, transceivers also have responsibility for receiving network signals. They are responsible for detecting packet collisions. When a transceiver on the transmit side detects a collision, its NIC link management function turns on a collision detect signal. The transmit side of link management in turn transmits a bit sequence known as a *jam*. This jam signal causes transmitting workstations to terminate their transmission and their NICs randomly schedule their retransmission. Meanwhile NICs on receiving workstations have been examining damaged packets, noting their fragmentary size, and discarding these *runts*.

Ethernet transceivers internally generate a signal known as a *signal quality error* (SQE) which is often referred to as a "heartbeat." This heartbeat is read by the NIC to ensure that the transceiver is working properly, sort of like having your spouse say "uh-huh" every couple of minutes so you know he or she is alive and listening to your story. Unfortunately, the heartbeats for 802.3 and Ethernet transceivers use different timing. Some manufacturers provide transceivers that can be manually set for one standard or another.

Because Ethernet broadcasts in all directions, it has cable length limitations as discussed earlier in this report. Original Ethernet with its thick coaxial cabling permitted one 500 meter segment with a maximum of three segments joined together by repeaters that rebroadcast the packets on the network.

Other limitations for 10Base5 due to signaling limitations include a maximum of 100 transceivers for a single cable segment with transceivers spaced at least 2.5 meters apart. The 10Base5 specification described drop cables connecting the workstation's NIC to transceivers on the bus. A break in a drop cable would not bring down the entire network, only the affected workstation. On the new "cheapernet" networks, however, the transceivers are built into the NIC. A break in cabling does bring down the entire network.

Besides cabling breaks, other common equipment-related Ethernet problems include *jabber* and individual station failures. A malfunctioning transceiver can jabber

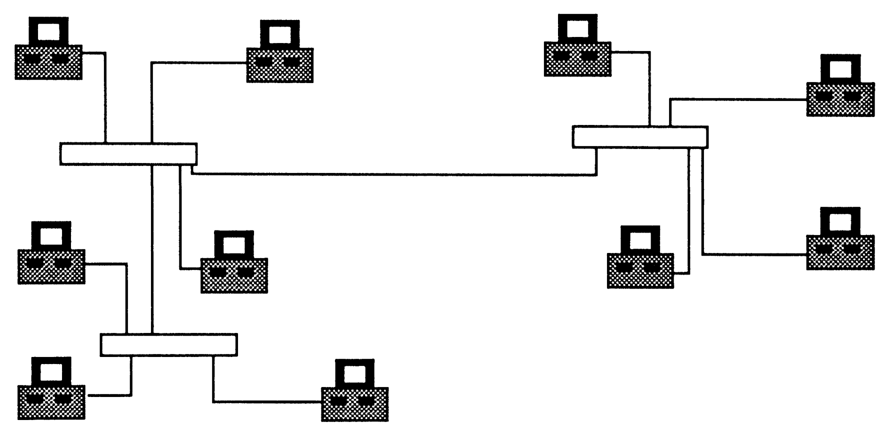


Figure 4.
An 802.3 Distributed Star Network

by sending out continuous streams of packets. This problem can be identified using a protocol analyzer. Similarly a malfunctioning workstation that sends packets below the minimum accepted length (runts) or packets that are continually maximum size with no variation whatsoever (causing traffic congestion) can also be spotted with a protocol analyzer.

A Bus By Any Other Name . . .

As you read earlier in this report, Ethernet and 802.3 are bus networks. The bus is a data highway that is laid in straight sections known as segments. Segments can be linked together, and repeaters can be used to extend a network's size.

The StarLAN topology and that of the new 10BaseT is a distributed star. Physically, as seen in Figure 4, such networks have workstation cabling radiating from concentrators in a star-like arrangement. Logically, though, they still operate as bus networks.

Sometimes known as *passive stars*, these networks enjoy the benefits of a star-like architecture because it is possible to maintain better network management and control. *Bypass circuitry* means that if one workstation fails, it can be bypassed by other workstations in that star by the concentrator or multiport transceiver from which the workstations' cables radiate.

10BaseT

Since you briefly reviewed at the new IEEE 802.3 10BaseT standard for Ethernet running on unshielded twisted-pair wire, this is a good spot to summarize the major specifications that make this product so appealing to systems integrators. Synoptics Communications was the first company to release an Ethernet product running 10 Mbs over unshielded twisted pair in 1987. It was followed by several other companies including David Systems, Hewlett-Packard, 3Com, and DEC. Unfortunately, each company had its own proprietary set of specifications that were incompatible with other companies' offerings. Large companies were reluctant to invest heavily in a small company's proprietary technology, because it very well might lead to a dead end should the small company fail. The 802.3 committee ameliorated these fears with its 10BaseT standard.

10BaseT has a 100 meter maximum segment length, a link integrity test specification, and the ability to disconnect a segment if there is a failure without bringing down the entire network.

A 10BaseT description of the Media Access Unit (MAU) indicates several new features not found in other 802.3 specifications. It has a jabber function that disables the MAU if it transmits after an allowable maximum time period. After being "gagged" for a period of time, the MAU will enable itself and try talking to the network once again. A workstation signal quality error (SQE) test discussed in this report monitors the health of each MAU and ensures that workstations are able to participate as receivers and senders of network packets. Repeaters under 10BaseT are able to disconnect a malfunctioning MAU without disconnecting all other workstations from using the network. When the failure is corrected, the specified port that has been taken out of service is reconnected to the network.

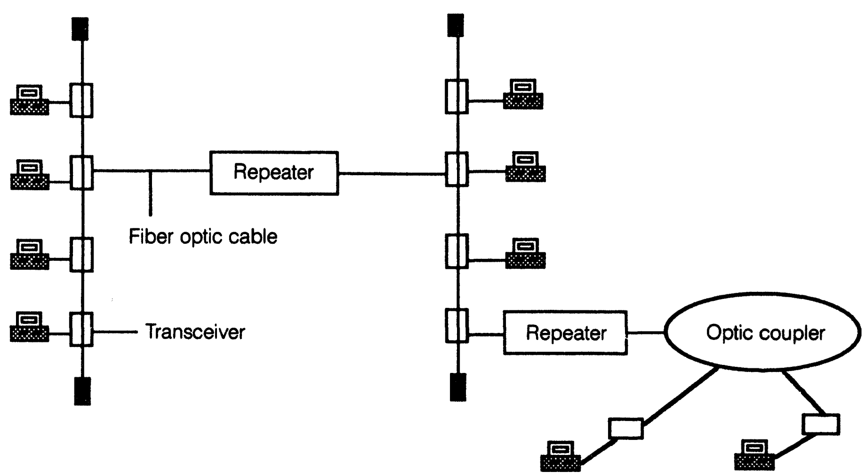
Still another useful 10BaseT feature is "intelligent squelch." This means that 10BaseT is able to function in an environment with a wide range of conflicting signals including voice traffic, the new Integrated Services Digital Network (ISDN) traffic, and asynchronous data traffic. Like a parent who is able to pick out his or her baby's cry even in a crowded room full of crying babies, intelligent squelch filters out other signals so that the Ethernet signals can be detected.

Another problem that 10BaseT was forced to address is the distortion that results to a signal as it travels over the twisted-pair wire. This problem long prevented high speed Ethernet traffic over this medium. A pre-equalization technique is used in which the signal is distorted before it travels to compensate for the distortion that will take place during transmission. The signal reaches its destination in very much the same form as it began before its initial distortion.

The major advantages of 10BaseT is that so much unshielded twisted-pair wire is already installed and that installation is easier and connections more reliable than the coaxial cable version of Ethernet. One large aerospace company recently linked 500 workstations on 10BaseT; it reported less downtime and easier installation than coaxial connections.

Having extolled all the virtues of 10BaseT, it is time for a friendly word of warning for the systems integrator who becomes a bit too optimistic about this "easy-to-work-with" medium. Buildings that appear to have lots of extra unshielded twisted pair can have poor assignment records so that it is almost impossible to trace back all pair. Pairs tied together somewhere inside the building might have

Figure 5.
An Optic Fiber Ethernet Network



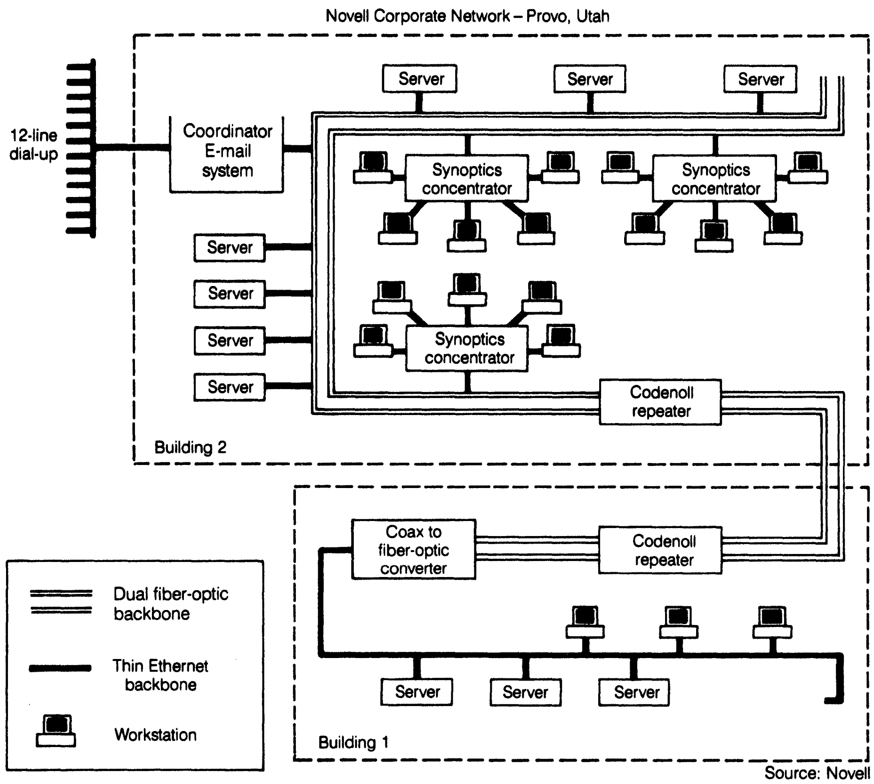


Figure 6.
Novell Networks Its Headquarters With Fiber Optic Ethernet Backbones

bridge taps. Some loops might hang in air, roads that lead nowhere. Conduits might be blocked, and pairs broken or poorly spliced together. Some systems integrators who are on the "bleeding edge" of this new technology have reported having to pull new twisted pair where often it initially did not seem necessary.

Ethernet on Optic-Fiber Cable

It is possible to use optic-fiber cabling on an 802.3 network. The major advantages of this approach are the immunity to any kind of electrical interference and the distance you can cover. The fiber-optic links can be up to 4.5 Km in length. Codenoll, one of the leading vendors in this market, points to its installation of the world's largest fiber-optic network, the 1.5 million square foot, 44-story network with over 3,000 stations on 92 miles of fiber at Southwestern Bell's headquarters in St. Louis, Missouri.

Each network workstation must have an NIC designed for 802.3 transmission over an optic-fiber network. Codenoll offers an external transceiver alternative. In any event, the transmitters in these products convert electrical signals to pulses of light while the receivers convert the lightwave signals back into electrical signals.

An optical bus star coupler sends optical signals to each station on a network. They are the equivalent of the concentrator or hubs you saw on the 10BaseT standard. Repeaters extend distances; they also permit *cascading stars* by connecting together different optic star couplers. Using Codenoll's products as an example, these couplers are available in coax-to-fiber, fiber-to-fiber, and coax-to-coax models. The actual fiber cables come with connectors preattached and replace coaxial cable and twisted pair wiring. Figure 5 illustrates an optic-fiber Ethernet network.

Using Optic-Fiber Backbones to Link Ethernet Networks

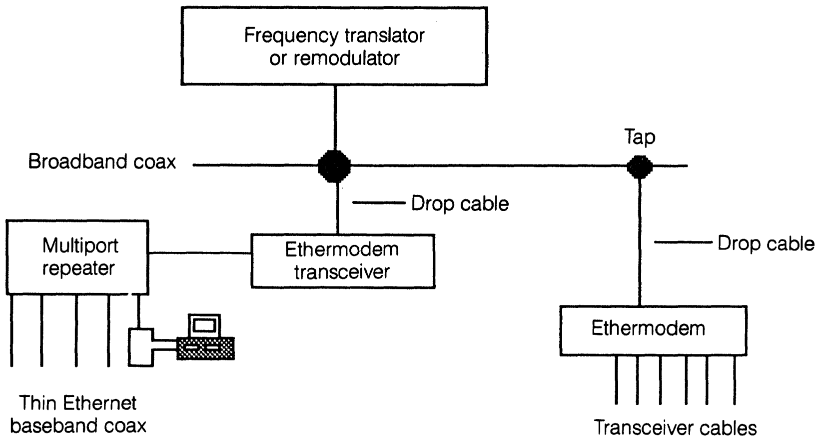
One popular use of optic fiber is as a backbone from which radiate concentrators and their networked workstations. Figure 6 illustrates how Novell in 1988 used this concept to link together two buildings. Building 1 at Novell's corporate headquarters was running a version of Arcnet while Building 2 was wired for Ethernet over twisted-pair wire. Because this installation occurred well before any 10BaseT standard had been adopted, Novell went with the leading twisted-pair Ethernet vendor at the time, Synoptics.

Notice that the Arcnet network in Building 1 contains a coax to Fiber-Optic converter. Once the packets are in optic form, a fiber-optic repeater transmits the packets to Building 2 where they travel along the fiber-optic backbone.

Ethernet on Broadband Coaxial Cable

The ability of Ethernet to run on broadband coaxial cabling is fairly recent. In September 1985, the IEEE standards board met and approved *10Broad36*, a broadband coaxial cable version of an 802.3 network that transmits at 10 megabit/second broadband coaxial cable standard with a maximum cable length of 3600 meters. What makes 10Broad36 so appealing to many companies is that it is now possible to support other local area networks such as token-ring on different frequencies while sending video and voice signals on their own frequencies. Each workstation's network adapter card contains a radio frequency (RF) transceiver that uses its modem to modulate the Ethernet signals onto their own network frequency or channel. This network can cover a distance consisting of approximately 4 miles. Broadband LANs use standard off-the-shelf cable television components, including a 75-ohm coaxial cable. Endpoints are terminated with a 75-ohm terminator.

Figure 7.
A Broadband Ethernet Network



Broadband networks use a technique called frequency division multiplexing to send information over channels, each of which has a designated frequency. Guardbands, frequencies that do not carry information, separate the frequencies carrying information and ensure that the separate channels do not interfere with each other.

Because we are talking about frequencies, we are talking about analog rather than digital transmission. Broadband networks use modems to convert data back and forth between digital and analog form. Assume that workstation 44 wants to send information to workstation 66. In effect, using a tree-like structure illustrated in Figure 7, workstation 44 on this broadband network sends its packet to a unit known as the *headend* using a specified frequency. The headend, acting like a radio station, retransmits this information on a different frequency to the destination workstation 66.

An Example of Broadband Ethernet

Boeing Commercial Airplanes has a complex multivendor network. It uses broadband coaxial cabling between buildings and in its factories and Type One shielded twisted-pair wiring within buildings to connect workstations using Synoptics' LattisNet version of Ethernet. The broadband cabling is divided into several forward and reverse 6 MHz channels with guard bands separating them.

Networked Apple Macintosh computers are linked to the broadband Ethernet network over a Kinetics gateway, which has a link into the Synoptics LattisNet concentrator. An Ethernet drop cable runs from the Fastpath gateway to a LattisNet transceiver and then to a spare port on the LattisNet concentrator.

Note that gateways can link the Ethernet network to IBM mainframes as well as DEC computers. TCP/IP plays an important role in this network. It provides a common denominator, a common set of higher level protocols for the DEC VAX, IBM hosts, Macintosh computers, and PCs running on Ethernet to share.

Why Use Ethernet or 802.3

Contrary to what you might hear from IBM, the entire world is not rushing to adopt its Token-Ring Network. Ethernet has several advantages. As the most mature network technology, it has the broadest vendor support and the largest installed base, which translates for the systems integrator into significant cost savings. Its media flexibility means that it is possible to mix and match different media (coax, twisted pair, fiber optic, etc.) as needed, which lowers costs still further. The new 10BaseT standard ensures that Ethernet will be around for quite awhile longer because its value/price ratio is so favorable.

The IEEE 802.4 Token Bus

Not all bus networks are the same. The 802.4 committee developed a token-bus arrangement based on broadband transmission. This section takes a close look at this type of network.

Description

IEEE 802.4 specifies a bus network similar to IEEE 802.3's bus topology, but one in which only one workstation tries to use the network at a time. This is a *noncontention* network because workstations are not actually contending or fighting among themselves for network access. Even though the network is cabled in a bus topology, the bus forms a logical ring in which each workstation is assigned a logical position. The workstation assigned the very last position is followed by the workstation assigned the first position. Each workstation is aware of the logical workstation following it and proceeding it. Interestingly enough, these are the only two addresses a workstation really knows. When you think of it, it is much the same approach used by espionage rings that are organized into groups of three. By knowing which workstation hands it the token and by also knowing which workstation receives the token from it, a workstation is able to identify a network breakdown (where is my token?).

The concept of a token is worth taking a moment to explain. Imagine, if you will, a campfire around which several Native Americans are sitting. A chief rises and raises a carved piece of wood that stands as a symbol of his or her authority, his or her right to speak. All the other speakers fall silent as he or she begins. When he or she finishes, he or she simply passes his or her symbol of authority to the next brave who then rises to speak. A control packet with a certain bit pattern known as a *token* operates in much the same way. Only the workstation controlling the token has permission to speak, and all other workstations must wait their turn to control this packet.

A workstation with the token has it for a specified period of time during which it may transmit one or more frames of information and poll other stations for their responses. A workstation that has very important network functions can be assigned additional time frames in a table that allocates this. There are actually four different priorities of access with the highest level (level 6) receiving the maximum amount of network bandwidth for its highest

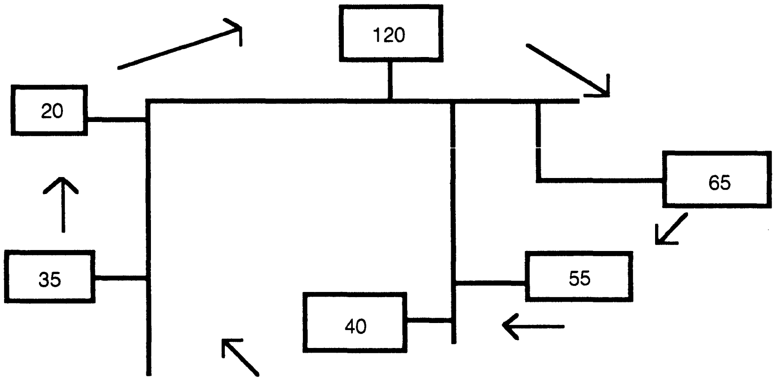


Figure 8.
A Token Bus Network

priority transmissions. Data can be classified into levels 6, 4, 2, and 0 priorities. After a workstation has broadcast its level 6 data and no other workstation has level 6 data to broadcast, the workstation may broadcast its level 4 data. The same principle holds for broadcasting level 2 data or even level 0 data. Each workstation controlling the token also has a responsibility to permit new workstations to join the logical ring. During its time period with the token, a workstation issues a solicit-successor frame. A *solicit-successor* frame is an invitation for a workstation to join the network at a location between itself and the next node in the logical ring. If there is no response to this invitation, the token controlling workstation passes the packet to the next workstation in its logical ring. If a workstation does request entrance, then the token is passed to it so that it can set all its parameters and linkages before the next workstation in the logical ring gets its chance.

As Figure 8 illustrates, a workstation with the token broadcasts its message over the entire network. Only the workstation with a destination address matching that of the packet will copy the packet to its NIC and process it. If a workstation is inoperative, it will time out and the token will be passed to the next workstation. While Figure 8 shows the token moving from higher addressed workstations to lower addressed workstations, they need not be adjacent to each other because you are looking at a logical bus and not a physical bus.

IEEE 802.4 also specifies that its packets travel along a broadband medium consisting of 75 ohm CATV coaxial cable. This means that several different channels can be transmitting different types of information on different frequencies simultaneously. Broadband also offers a couple of other major advantages over the baseband cabling associated with Ethernet and the IEEE 802.3 standard. Taps can be placed anywhere on a broadband network while Ethernet's baseband standard usually requires a minimum distance of 2½ meters between nodes. In addition to broadband's less sensitivity to noise, it can stretch much further than baseband. With repeaters a broadband network can extend up to 25 miles. Depending upon the number of channels used, data rates between 1 Mbit/second and 10 Mbit/second are possible.

The 802.4 Frame

Figure 9 illustrates the 802.4 frame; it is actually a bit more complicated than it looks. At first glance, it looks very similar to an 802.3 frame. The Preamble is a bit pattern that enables workstations to synchronize their signals. The Start Delimiter indicates the start of a frame. The Control field can indicate a number of different formats. It can function as a token, for example. A different bit pattern means that the frame is functioning as a "Who follows" frame. This is a request to other workstations to try to match the Source Address and reply which workstation should next receive the token.

Still another bit pattern in this Control field indicates that the data field is carrying meaningful data. So, the Control field really indicates whether the entire frame is to be interpreted as a control message and/or request for information or whether it is to be interpreted as meaningful data to be delivered.

The Destination and Source Address fields are self-explanatory. They are particularly significant in 802.4 because they are used to help determine the successor or next workstation to use the token in situations where a workstation fails to respond. Remember, the workstation failing to respond knows who it is supposed to receive the token from. It can match that address with the Source Address in a frame that is carrying a "Who follows?" control request.

The Frame Check Sequence field provides a 32-bit cyclic redundancy check on all fields except the Preamble, Starting and Ending Delimiters, and itself. Finally, the End Delimiter indicates the end of a frame. If an error is detected by a receiving workstation, it will set an error bit in this field to one that indicates that the source workstation needs to retransmit the frame.

Some Advantages of a Token-Bus Network

One advantage of a Token-Bus network is that each workstation is guaranteed a chance to access the network no matter how many workstations or how much traffic. When a station passes the token, it starts a timer that runs until the station is ready to pass the token again; in other words, the timer measures a complete rotation around the ring of

Preamble	Start frame delimiter	Frame control	Destination address	Source address	Data	Frame check sequence	End frame delimiter

Figure 9.
A Token Bus Frame

Figure 10.
MAP's Set of Protocols and the OSI Model

Application Layer	Manufacturing Message Specifications File Transfer Access & Management (FTAM) X.500 Global Directory Network Management Agent Associated Services and Control Elements
Presentation Layer	Presentation
Session Layer	Basic Combined Subset
Transport Layer	Transport Class 4
Network Layer	Internet Protocol
Data Link Layer	802.2 Logical Link Control 802.4 Media Access Control
Physical Layer	10 Mbs Broadband 5 Mbs Carrier Band

the token. The workstation now looks to see if it has anything else to transmit using a system that prioritizes messages to ensure that the highest prioritized messages always are transmitted first.

Eventually a workstation's token time will exceed a maximum set by the network administrator. The workstation will then pass the token to the workstation it has listed as the next destination for token. Another advantage to Token Bus is its broadband media, which makes it less vulnerable to electrical interference and thus a better choice for factory communications.

Finally, the priority system on a Token-Bus network ensures that highest priority messages will always be transmitted. A workstation must transmit its highest priority items (THT) before it is permitted to transmit its next level of priority (TR4).

Protocols to Manage the Factory and Office

Several years ago, General Motors realized that it was facing an automation nightmare unless it took immediate action. Its factory floors were filled with different types of computerized tools including several kinds of robots; unfortunately, these various devices were unable to communicate with each other and unable to send information to other parts of the plant. The company began developing *Manufacturing Automation Protocol* (MAP), a suite of protocols based on the OSI model that would facilitate network communications with special provisions for a manufacturing environment.

Because a factory floor is vast, often well beyond the scope of baseband coaxial cabling, and because of the amount of potential electrical interference, a broadband token-bus network seemed the logical LAN choice. GM worked closely with a powerful MAP users group to develop the protocols needed.

At approximately the same time that General Motors was wrestling with its automation nightmare, Boeing Computer Services set out to develop a suite of protocols to cover the office environment and address its very different needs. *Technical and Office Protocol* (TOP) was developed specifically for office services such as engineering, marketing, accounting, and administration that supported a manufacturing environment such as that found at Boeing. TOP is also a subset of the OSI model's set of protocols, but an 802.3 bus network was selected as its topology.

While 802.3 and 802.4 have different frames and different network access methods as discussed earlier in this report, they share the same Logical Link Control data link protocol. Because they share the same LLC, it is possible to build bridges connecting MAP and TOP with each other as well as with 802.5 networks. Both Ethernet versions 1 and 2 have a different LLC, which would have created a serious incompatibility issue.

Figures 10 and 11 reveal the protocols associated with MAP and TOP versions 3.0. Notice that both sets of protocols are identical from layers 2 through 7. The major difference, beside the 802.3 versus 802.4 split, was in the Application layer protocols. Because MAP and TOP have very different user needs, it is precisely in this layer that you would expect to see differences.

Because MAP was developed specifically to meet the needs of manufacturing, its Application layer contains Manufacturing Message Standard (MMS), an OSI Application layer protocol designed for formatting and transmitting commands between controlling programs and machines. MAP 3.0's version includes the ability to support robots and other real-time control applications.

MAP contains the Association Control Service Element (ACSE) and Remote Operations Service Element (ROSE); these protocols enable MAP to start jobs on remote systems and bind them into communications with the parent application.

MAP Network management protocols gather information on the usage of the network media by network devices and ensure the correct operation of the network as well as provide reports.

TOP, on the other hand, places more emphasis on office automation needs. It uses the OSI X.400 standard for electronic mail, the emerging X.500 standard for global directories, and the VT protocol for terminal emulation. TOP also contains protocols for graphic data interchange and

Figure 11.
TOP's Set of Protocols

Application	X.400 X.500 FTAM Association Control Service Element
Presentation	OSI Presentation Protocol
Session	OSI Session Protocol
Transport	OSI Transport Protocol Class 4
Network	OSI Connectionless Network Protocol
Data Link	LLC Type 1
Physical	IEEE 802.3 IEEE 802.5

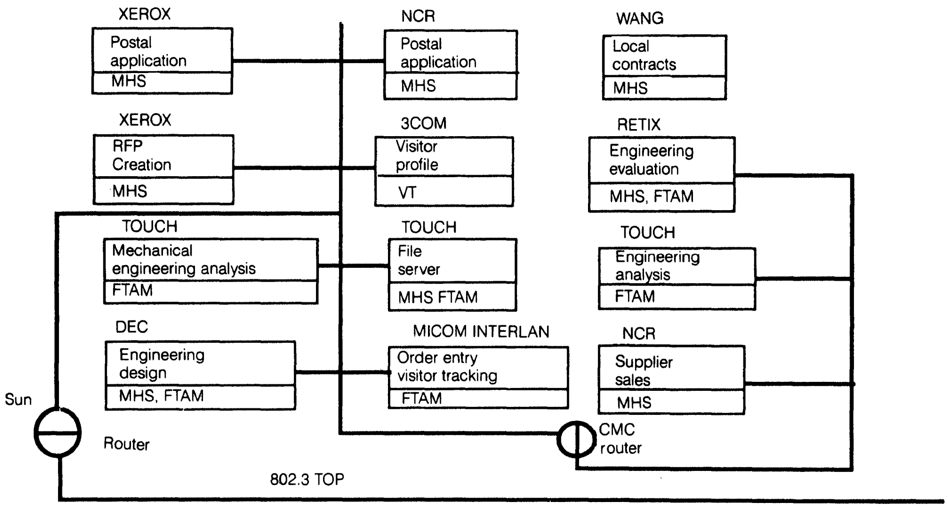


Figure 12.
TOP's Set of Protocols

distributed transaction processing. MAP/TOP protocols are compatible with the Government OSI Profile (GOSIP).

The Corporation for Open Systems (COS) has a special program that certifies products that meet its standards for compatibility with OSI's protocol stack. It offers an integrated tool set that tests products for MAP/TOP 3.0 compliance.

While General Motors helped nurse MAP and Boeing developed TOP, both corporations have turned over leadership of these standards to a joint MAP/TOP Users group. Recently leadership has passed to a permanent independent organization called the Information Technology Requirements Council (IRTC).

MAP and TOP in Action

A good example of MAP in action is General Motors' assembly plant in East Pontiac, Michigan. Robots provide 98% of all welds while a vision system measures 40 dimensions on each truck. Thirty-two miles of cabling support 2,000 devices that can transmit data over a MAP network at 10 Mbs. What makes this plant so important in supporting the MAP cause is that in addition to the hundreds of different types of robots and process control machinery, the plant has a wide range of different vendor computer systems including IBM hosts and DEC VAXes. What makes communication possible and efficient is that all machinery share the same suite of MAP protocols.

The 1988 Enterprise Event and the Freezing of MAP/TOP Standards

In June 1988, a convention known as the Enterprise Networking Event took place at the Baltimore Convention Center. Vendors demonstrated real products running under MAP/TOP. The theme of this event was the Enterprise-wide network, a network that includes both the factory and the office. General Motors' demonstration featured two robots that worked in unison to assemble an enterprise product according to visitors' specifications. The booth also featured electronic mail applications and use of the MAP/TOP Network Manager and Directory Server protocols.

The point of this convention was to demonstrate products from multiple vendors that worked together because of their MAP/TOP compatibility. Boeing's booth featured eleven different vendors. Visitors entered personal information into a visitor tracking system that demonstrated order entry and bar code reading. The bar code was read and compared against the visitor's personal files to verify registration.

The Boeing booth also exhibited several aspects of the procurement process including creating a request to be sent to suppliers, selecting the suppliers, preparing and delivering the request electronically, receiving and evaluating the responses electronically, sending the order, and then finally receiving the order acknowledgement from the supplier. Figure 12 illustrates how this network was organized. The backbone ran TOP protocols over IEEE 802.3 baseband coaxial cabling. Notice that the OSI FTAM and VT protocols that TOP has incorporated for file management and terminal emulation were used.

Presently MAP implementations are growing much faster than TOP installations. Many vendors and potential customers were very disturbed when MAP was upgraded from version 2. 1 to 3.0 because of the number of changes that were not upward compatible. Why spend money developing a product or implementing MAP protocols when the standard seems so unstable? The MAP/TOP group responded to this criticism at the Enterprise Event by announcing that MAP will not be changed for six years. This meant that companies count on a stable period extending to 1994. Many industry experts feel this announcement will spur additional MAP adoptions.

MAP and TOP in the Future

If a recent meeting of the North American MAP/TOP Users Group is any indication, specifications for both these products will be broadened in the near future to include several OSI as well as other protocols. There appears to be a good deal of support for an 802.4 fiber-optic version as well as a Fiber Distributed Data Interface (FDDI) and Integrated Services Digital Network (ISDN) versions.

DEC's loud pronouncements that 802.3 should be included under MAP has sparked controversy for the past couple of years. Because DEC is a major player in the factory automation market, it is not surprising that they have pushed so hard for this addition. Hewlett-Packard recently announced a MAP product that supports 802.3.

While General Motors has dropped its leadership role and let the users group administer MAP/TOP, a spokesperson has announced the company would oppose 802.3's addition to MAP.

Summarization of Ethernet Concepts

Ethernet was the first nonproprietary local area network, and it still commands the largest number of users. The IEEE 802.3 standard for a CSMA/CD contention bus network is similar to Ethernet, with some significant differences. 10BaseT is a new 802.3 standard for unshielded twisted-pair transmission at 10 Mbs. 10Broad36 represents the broadband coaxial cable version of the 802.3 standard.

The 802.4 standard for a Token-Bus network describes a noncontention network which is a physical bus but a logical ring. Manufacturing Automation Protocol (MAP) is based on 802.4. Technical and Office Protocol (TOP) is a suite of protocols closely based on the OSI model and 802.3 that provides network services for such departments as accounting, marketing and engineering as they support manufacturing and its MAP protocols.

The IEEE 802.5 Token-Ring

The IEEE 802 committee developed a standard which became known as IEEE 802.5 for a noncontention local area network, a network that is a logical ring but a physical star. Cables to individual workstations radiate from wire concentrators known as Multistation Access Units. Workstations transmit information in a packet known as a *frame.* IEEE 802.5 committee published its set of specifications in a Blue Book and has been adding to this standard on a regular basis. The original specification described a 4 Mbs transmission speed. IBM and several other vendors now offer a 16 Mbs version, which is covered later in this report.

The Open Token Foundation and 802.5 Interoperability

If you have ever worked with LANs, you are aware that IEEE 802 specifications by themselves do not guarantee interoperability since they represent guidelines. IBM, for example, has published programs that meet 802.5 specifications but contain calls directly to their own NIC firmware; the result is that the programs do not work with other vendor 802.5 products. Several vendors would like to see a movement toward greater cooperation between the IEEE and 802.5 network equipment manufacturers and increased interoperability among 802.5 products.

In 1988 a group called the *Open Token Foundation* (OTF) was formed to increase 802.5 network interoperability. The OTF has staged a demonstration at NetWorld in Dallas in 1989 where more than 20 companies demonstrated their products' 802.5 interoperability.

Take this interoperability issue a step further. For two network nodes to communicate with each other, they must run the same higher level protocols so that they can "understand" each other. The same two network nodes can coexist even with different higher level protocols and hardware as long as they share the same 802.5 Medium Access Control (MAC) layer. It is this layer that enables the nodes to recognize which frames are to be processed or dropped and to interpret network addresses correctly. The IEEE 802.5 committee has been involved in coexistence issues while the OTF is concerned with both coexistence and communications issues. Table 1 indicates how differences in token-ring elements affect the ability to coexist or communicate.

Table 1. Coexistence and Communication

These Token Ring Elements:	Need to Be the Same Between Nodes for:	
	Coexistence	Communication
Physical layer signaling	Yes	Yes
MAC layer and frames	Yes	Yes
Address and network management	Yes	Yes
Source routing header filtering	Yes	Yes
LLC layer SAP address filtering	Yes	Yes
LLC types (I, II, or III)	No	Yes
Upper layer protocol (TCP/IP NetBIOS)	No	Yes
Card/PC hardware interface	No	No
System processor (80XXX, 68XXX)	No	No
PC machine type	No	No

Source: Data Communications

The Use of a Token on 802.5 Networks

When you examined the IEEE 802.4 token-bus standard earlier in this report, you saw a token consisting of a predetermined bit pattern can only be used by one workstation or network node at a time. The transmitting workstation physically alters this token's bit pattern, which announces to all other workstations that this token is being used. This method resembles the way a taxi cab driver alerts potential customers that his or her cab is in use by setting an "in use" flag on top of the cab. The token, now transformed into a frame of information containing the message to be sent is transmitted around the ring until it reaches its destination workstation.

Messages sent along a token-ring network are received by each network workstation which in turn checks to see if it is the frame's correct destination. If it is not, the workstation acting as a network repeater retransmits the frame to the next network workstation. Finally, the destination workstation receives the message and copies it to its internal memory before retransmitting the frame back to the sending workstation. The frame makes its way around the ring back to the sending workstation, which observes that the message was successfully copied and then resets the token so that it is available for another station to use.

This passing of a token might remind you of a children's game in which children sitting around a circle transmit a message by rebroadcasting the information by whispering it from one child to another until the child who started the cycle hears the information. What happens if a child falls asleep and cannot transmit the message? Obviously, the message would fail to come full circle. Later in this report you will see how a multistation access unit or wire concentrator alleviates this problem by bypassing inactive workstations so that the continuity of the ring is not broken.

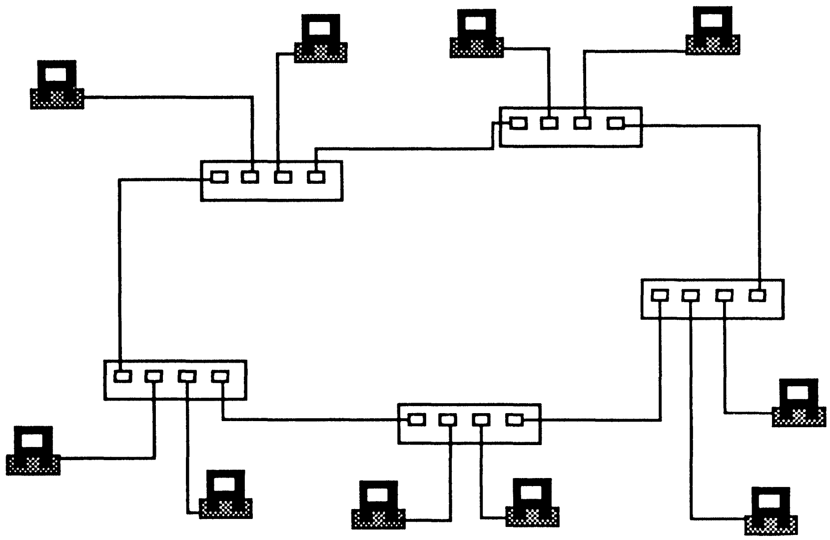


Figure 13.
Multistation Access Units in a Token-Ring Network

Basic Components of a Token-Ring Network

This section looks at the key components of a token-ring network including its network interface card, its multistation access unit, and its cabling. It is important to recognize that IBM's Token-Ring Network (IBM loves to use capitals as well as abbreviations) differs in some ways from other 802.5 networks. That does not necessarily mean IBM's approach is better, but it does mean that you have to be careful not to mix and match NICs with different chip sets or different 802.5 cabling schemes.

Network Interface Cards (NICs)

Workstations must have Token-Ring network interface cards (NICs). These cards contain the Medium Access Layer protocols that are at the heart of the 802.5 standard. Not all 802.5 NICs are the same; while all 802.5 standard NICs support IBM's Source Routing scheme, IBM's NICs actually handle the Source Routing with their firmware and do not require additional software to perform this task.

IBM and Texas Instruments jointly developed an 802.5 chip set to ensure that third-party manufacturers of LAN equipment would provide compatible products. Ironically, virtually all 802.5 NIC manufacturers use this chip set except Ungermann Bass and IBM. Even though dozens of vendors offer 802.5 NICs, there are significant differences. One major difference is in the way these cards handle memory.

The three major approaches toward handling memory are *Shared Memory*, *Direct Memory Access* (DMA), and *Bus Mastering.* With Shared memory, a portion of the host's memory is mapped to the NIC's memory. The host can read information directly from its own memory, thus providing very fast access. The NIC will have jumper or switch settings to assign the buffers on the card to a system memory address. The CPU reads the RAM information on the card just as if the information was found in high memory RAM on the motherboard. There is a drawback, however. How much host memory does the manufacturer allocate? Too little memory allocation will result in the need for additional communications and transmissions that will slow down operations. Also, some information could be lost during this process. Too much memory allocation could cause conflicts with other key operations. The normal address range used by the chip set for this function could create conflicts with other functions that use this range of memory including system video memory and the Extended Memory Specifications (EMS) paging area.

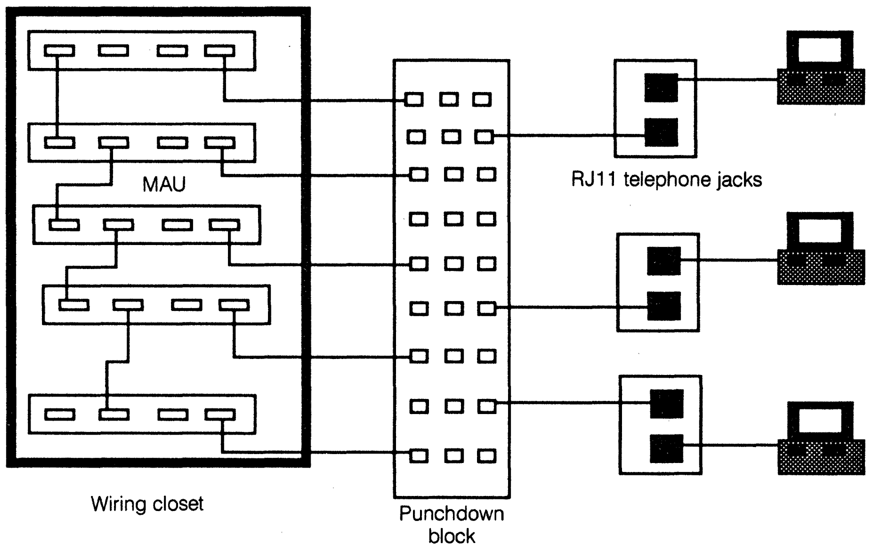
Direct Memory Access (DMA) offers an alternative memory approach. A DMA controller residing on the NIC's motherboard assumes responsibility for determining the source and destination addresses of the data to be moved along the data bus. It requests and receives permission to use the data bus and then performs the necessary read and write operations. Unfortunately, on PCs the DMA speed remains what it was on the original Intel 8088 machines, 4.77 MHz. This approach is not as fast or efficient as Shared Memory. It is reasonably efficient when used for large data transfers, but very inefficient because of overhead when used for small data transfers. As a systems integrator, your ability to determine what kinds of data transfers will take place routinely will help you decide whether this DMA approach is acceptable for the network you configure.

The third memory approach, Bus Mastering, is the most efficient way to use memory, but you need the Micro Channel Architecture (MCA) based PS/2 workstations or the Extended Industry Standard Architecture (EISA) 16-bit bus clones and appropriate NICs use it. The NIC in effect is the "Master of the Bus" because it controls the bus and not a DMA controller sitting on a motherboard. The NIC writes information directly to the host's memory without the need for request permission. Also, the NIC only needs to perform a write operation and not the read operation required with the DMA approach. If the network you are configuring has a lot of small data transfers, this approach might be significantly faster for you than the DMA approach.

Just as NICs use different memory schemes, they also offer different sized buffers. It is particularly important that you select a card with a very large buffer for your file server so it can handle its heavy I/O load.

Other significant features that differentiate 802.5 NICs include the software included on them and the quality of their driver software. Demand Protocol Architecture (DPA) is 3Com's approach to minimizing RAM cram by using special software to load protocols as needed. Similarly, some 802.5 NICs include protocols as well as other

Figure 14.
MAUs in a Wiring Closet Connected to Workstations Using a Punch-Down Block



software so that workstations need not try to cram them into RAM. IBM's Token-Ring NICs include the Logical Link Control (LLC) 802.5 software.

Take software included on an NIC a step further. TI's 802.5 chip set includes basic driver software, but some companies have rewritten these drivers for their NICs using assembly language routines that speed up the throughput between the operating system and the hardware.

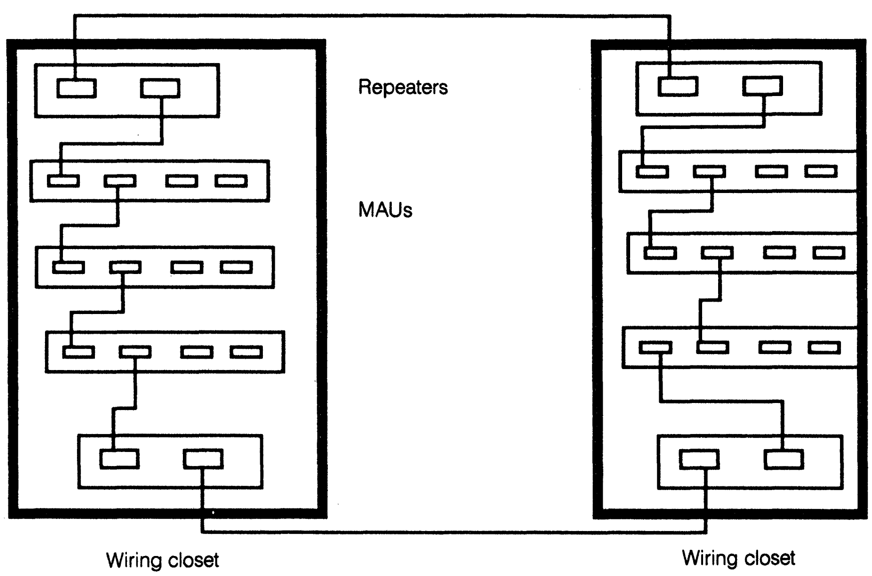
I have friends who never thought they'd live to see IBM salespeople suggesting a rival product, but up until LAN Server became a viable product, "Big Blue" salespeople were seen recommending NetWare to their large corporate accounts. By Novell's own account, over fifty percent of IBM Token-Ring Networks run NetWare. In any event IBM has opened up its networks to make it easier for systems integrators to mix and match some products. When it released the PC LAN program, the NetBIOS Extended User Interface (NetBEUI) was required as an interface to NetBIOS, IBM's de facto network interface standard and its equivalent to the OSI model's Session layer. IBM's own Token-Ring NICs used the TOKREUI adapter handler to interface with the NetBEUI. More recently, IBM introduced its LAN Support Program which replaces NetBEUI and permits installable device drivers. This means that a systems integrator now has more flexibility. It is possible, for example, to use 3Com's Token Link NICs with 3Com's 3ComREUI adapter handler as an interface to NetBIOS. The network could then run a wide range of network operating systems such as NetWare or VINES. The NetBIOS link could be handled by this 3Com device driver.

Cabling and Multistation Access Units

The cabling connecting the network can range from type 3 unshielded telephone wire to include optical fiber and IBM data grade cabling. The promised 4 megabit/second transmission speed might not be achieved using the unshielded telephone wire if there is any interference. Another limitation of using inexpensive telephone wire is that it will support a maximum of only 72 workstations compared to the 260 workstations that can be supported on one ring with data grade coaxial cabling.

While a Token-Ring network is a logical ring, it is a physical star with workstations radiating from *Multistation Access Units* (MAUs). These MAUs contain bypass circuitry so that if a workstation is not plugged in or logged off the network, a relay bypasses the workstation's MAU

Figure 15.
Repeaters Extend a Token-Ring Network



Starting delimiter	Control	Destination address	Source address	Information	FCS	EFS	Ending delimiter	Frame status

Figure 16.
The 802.5 Frame

port so that continuity of the ring is not broken. Figure 13 illustrates how these MAUs function.

Usually MAUs are placed in a centralized wiring closet. The cable connecting the MAU to a workstation is known as a *lobe*. The maximum distance of these lobes varies according to the type of cabling used with the 4 Mbs version of Token-Ring network supporting a maximum length of 330 feet for twisted-pair telephone wire and 984 feet for data grade media.

Figure 14 illustrates a group of MAUs in a centralized wiring closet. Connections at the wiring closet are made at punchdown blocks or through patch panels. Connections at the workstations are made using standard telephone wall jacks.

MAUs and Repeaters

For larger installations it might be impossible or simply undesirable to place all MAUs in one central wiring closet. Copper repeaters can extend the distance between MAUs by up to approximately 2,500 feet. Figure 15 illustrates how these repeaters are linked to MAUs.

Some rules of thumb are associated with token-ring networks. Generally, you want to minimize the number of wiring closets needed. With twisted-pair telephone wire, you can have a maximum of 72 workstations using up to 9 MAUs located in a single wiring closet as long as all lobe lengths are limited to 1,000 feet or less. If you go beyond this length, you will need to use repeaters or bridges.

With shielded data cabling (Types 1 or 2), a Token-Ring network can consist of a maximum of 260 workstations with up to 33 MAUs located in a single wiring closet with all lobe lengths limited to 2,000 feet or less.

The 802.5 Frame

The 802.5 frame is quite different from the 802.3 frame examined earlier in this report; this explains why you simply cannot plug the two networks together. Figure 16 illustrates the 802.5 frame.

The Starting Delimiter signifies the beginning of a frame; its unique pattern (JK0JK000) prevents it from being mistaken for data. J and K here represent non-data symbols. The Access Control field is where you find the priority and reservation bits as well as the monitor bit. A workstation can use any one of eight different priority levels to indicate that it needs to reserve use of a future token.

Figure 17.
An 802.5 Token's Format

Start delimiter	Access control PPPTMRRR	End delimiter

P Priority mode
T Token bit
M Monitor count
R Priority reservation

Other workstations will compare their priority levels with this number and defer if they see a higher priority in this field. The monitor field refers to token management, a topic considered shortly.

The Frame Control field is used to indicate whether the frame contains Logical Link Control (LLC) data or a Medium Access Control (MAC) control parameters. The Destination Address and Source Address fields are self-explanatory. The Source and Destination address fields are designed to convey a number of different kinds of information. The first bit is set to 1, then the message is a group broadcast for everyone on the network. An initial zero bit, on the other hand, indicates that this is a message addressed to a specific workstation. The second bit in the address field indicates whether the address is global (0) or local (1). *Local* refers to another node on the same network. Each workstation has a unique 48-bit address obtained by the manufacturer from IEEE and burned into the PC Adapter ROM chip. Locally administered addresses are assigned by the local network administrator; these addresses override the universally administered addresses found on the ROM chip. The address fields have been designed to accommodate the addresses of workstations that exist on other rings; in fact, the first two bytes of these two fields are designated for a workstation's ring number. The addressing can become quite complex because the 802.5 committee has proposed an address structure that includes space for indicating multiple rings, bridges, etc., very much like a letter to a college student might require several lines of address to indicate the specific college, street address, dormitory, post office box, etc.

The Frame Check Sequence field is used for error checking, while the End of Frame Sequence and Ending Delimiter are fields associated with specifying the end of a frame. The Frame Status field indicates whether the frame is **Good** or **With-Error**. If the frame is **With-Error**, the field specifies one of several reasons including an FCS that does not match, a frame too large for the buffer space, or a frame too small (*runt*) to be valid.

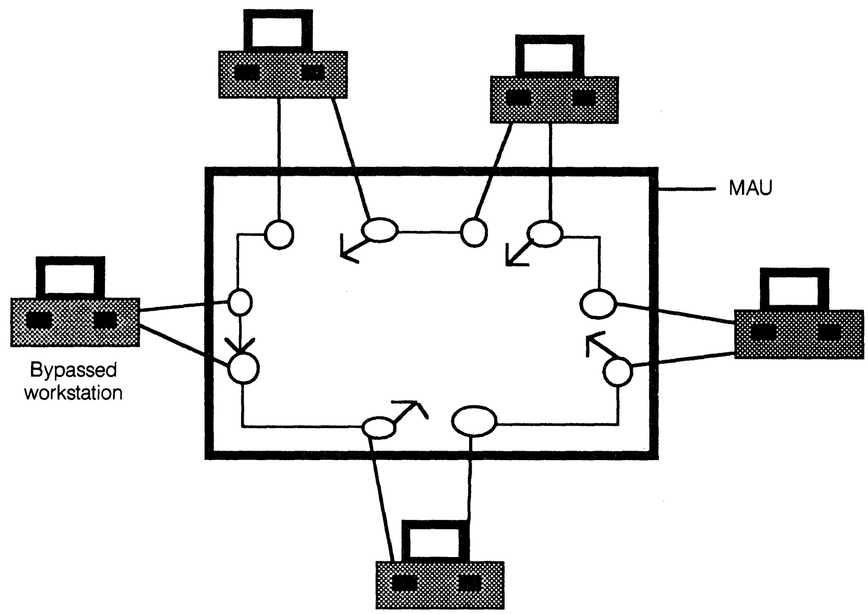
The Frame Status field is used by the workstation originating the information frame to determine if the workstation designated to receive the message actually recognized its own address. The originating workstation sets the Addressed Recognized bit to 0 (zero) while any other station on the ring sets it to 1 if it recognizes the address as its own. The originating workstation also sets the Frame Copied bit to 0. When the receiving workstation copies the frame into a read buffer, it sets this bit to a 1. When the frame returns to the originating workstation, it checks this key Frame Status field to see if the frame was recognized and copied correctly.

The Token on an 802.5 Network

A token is nothing more than a specific bit pattern that the workstations can recognize. Figure 17 illustrates the token on an 802.5 network. A workstation needing to use the network will grab the token when it arrives and change one bit to transform it from a token into what is known as a Start of Frame Sequence (SFS).

Workstations that need to use the token on an 802.5 network can indicate any one of eight different levels of

Figure 18.
A Token-Ring Network with a Defective Node



priority while placing a reservation for it. A network workstation releases its token after each transmission because it is not permitted to broadcast continuously regardless of its high priority. The monitor bit is broadcast as a zero in all frames and tokens except the monitor itself in order to ensure that no frame monopolizes the ring and no token has a priority greater than zero. Every workstation thus has equal access to the network. When a workstation is ready to continue a message's path through the network by rebroadcasting its bit pattern, it examines the reservation bits (RRR). If it has a message of its own that it wants to send and its priority is higher than the present sender, it raises the value of this three-bit field to its own level assuring that its message is the highest one waiting to be transmitted. Further down the network a second workstation with even a higher priority might change this bit pattern to reflect its own priority level and thus "bump" the previous workstation from its reservation for the next available token. The priority established in the PPP Priority field reflect the workstation's priority to use the network. If a workstation observes that its priority is higher than the one already reflected in the RRR Reservation field, it raises this value to its own level and thus reserves the next token for its own use. The address of the bumped workstation goes into a memory location that serves as a queuing area for displaced workstations seeking access to the network but forced to wait their turns.

The Role of the Active Monitor

Token-Ring Network uses an *Active Monitor* to ensure smooth network operations and handle error conditions. The first workstation to join a ring becomes the Monitor, sort of like being President of a country consisting of one citizen. As the Monitor, this workstation generates an Active Monitor Present frame periodically that tells other workstations that there is a Monitor alive and functioning. Other workstations issue Standby Monitor frames periodically to indicate that they are ready to assume this role if the Monitor fails and it becomes necessary.

What happens if the Monitor does fail? Each workstation keeps track of time. When one realizes that it is long past time for the Active Monitor Present frame, it generates a Claim Token frame. Other workstations begin to realize the same thing now, and they too begin to generate these Claim Token frames. Which workstation is going to win the job of Active Monitor?

Now something very interesting happens, sort of like the way dogs determine which one is going to rule the neighborhood. Each workstation receives Claim Token frames issued by other workstations. It compares its Source Address with the Source Address of the Claim Token frame it receives. If it has a lower Source Address, the workstation drops out of this contest and begins issuing Standby Monitor frames. If it has a higher Source Address, then it ignores this other contender's frames and continues generating its own Claim Token frames.

The Frame Check Sequence field is responsible for error checking for the Frame Check, Destination Address, Source Address, and Information fields. Bit seven of this field is the error-detected bit. The workstation originating a frame sets this bit to zero while the first station to detect a transmission error sets it to one. The first station to detect this error flag counts the error and prevents other stations from also logging this error. This method helps to localize where an error has taken place. The workstation serving as the Monitor scans the network for transient and permanent errors. Transient errors are logged by various workstations on the network as "soft error conditions." These errors generally can be corrected by retransmitting the frame. Permanent errors, on the other hand, can disrupt network operations. If a frame comes back with the indication that a destination workstation has not recognized its own address and copied the frame, the Monitor workstation can use the bypass circuitry built into the multistation access units to bypass the defective station and maintain the network's integrity.

Error Checking or "Someone to Watch Over Me"

The Active Monitor examines the frames circulating on the network and removes any of them that are defective, issues a new token, and ensures that the network runs smoothly. It also helps identify and remove defective network nodes. Figure 18 illustrates a Token Ring network with four workstations (1-4). Workstation 1 starts to receive fault messages indicating that a problem exists somewhere between it and its Nearest Active Upstream Neighbor (NAUN). Workstation 1 begins issuing Media Access Control (MAC) layer beacon frames that contain its own address as the source address with its NAUN (workstation 4) as the frame's destination address. When workstation 4 receives a total of eight beacon frames, it removes itself from the ring and begins self-testing. It is able to test its own NIC hardware, lob cabling and interface to the MAU. If workstation 4 does not find any errors, it will re-enter the ring. Note that when a workstation is powered on, the NIC initiates a self-test that checks its memory and circuitry. If an error is found, the workstation will not enter the ring until it is corrected.

Meanwhile Workstation 1 is still receiving error messages. It knows workstation 4 is not the guilty party? Horror of horrors, could it be the cause of these error messages? Workstation 1 takes itself out of service and begins self-testing. If it does not find anything wrong, it will return to the ring. Because Workstation 1 was the Active Monitor, the remaining workstations will contend for this job using the method described in the previous section.

The functions of the Monitor workstation are provided by the IBM Token-Ring Network Manager Program. It provides continuous monitoring of the network for errors as well as the corrective actions discussed here. It also logs network error and status information for report generation. It is this program that provides the mechanism for the creation of addresses for each workstation as well as the establishment of passwords.

Common Problems on Token-Ring Networks

When you examined Ethernet earlier in this report, you saw that there were a number of common network problems including cable breaks and "jabber" conditions. Many systems integrators will swear that even with all its self-diagnostics, Token Ring is far more difficult to maintain than Ethernet. 802.5 networks also have their cable breaks, and it is also possible for the relays that take a workstation out of service on an MAU to stick so that the integrity of the ring is broken. Sometimes NICs will garble or abbreviate frames. There are many reasons why this can happen including electrical interference, low batteries, NICs going bad, and loose connectors.

Sometimes on Token-Ring networks, a token is accidentally duplicated or lost. Two workstations might believe they hold the token and then attempt to transmit a frame simultaneously. This situation is handled very nicely by the network. Workstations examine the Source Address field of a returning frame and make sure that this address matches their own address. If the two addresses do not match, the station aborts its own transmission and does not issue a new token.

IBM's 16 Mbs Token-Ring Network

Today, several vendors including IBM offer 16 Mbs versions of their 802.5 networks. IBM offers a 16 Mbs version of its Token-Ring network requiring its own special adapter cards which come with 64K of on-board RAM. This extra memory allows larger frame sizes and more concurrent sessions when the adapter is used in a server. The larger frame sizes permit high volume RAM-to-RAM transmissions such as images.

This high-speed version of a token-ring network comes at a price, though, because it requires IBM Type 1, 2, or 9 shielded cabling and cannot use the inexpensive unshielded twisted-pair wire. Wire closets can be up to two kilometers apart when connected with fiber-optic cable. Although IBM claims that it cannot use unshielded media for 16 Mbs transmission speeds because of electrical interference, other companies such as Proteon (ProNET-10) offer speedy alternatives using this media. At the moment, many experts feel that the primary function of such speed is to serve as a *backbone network*, a kind of super router that connects several different networks together.

The Importance of Fiber Optics with 16 Mbs Token Ring

Because the companies likely to install 16 Mbs Token Ring are likely to be the companies with very large networks, fiber optics will play an important role in such networks' design. Many vendors offer optical-fiber converters to Token Ring. These converters permit optical links between wiring closets with 62.5/125 or 100/140 multimode optical-fiber cabling.

These relatively new fiber-optic systems often provide network management software as well as two distinct signal paths. One path is in the primary direction of the network while the other path is in the backup direction. Normally the primary path will carry the network traffic while the backup path carries ring maintenance signals. The optical converters are intelligent enough to remove themselves from the ring if they sense that they are malfunctioning and then later to reinsert themselves if they no longer identify a problem. Early Token Release on the 16 Mbs Token-Ring A workstation on a 16 Mbs Token-Ring Network is permitted to transmit a token immediately after sending a frame of data instead of waiting for its frame to return. Figure 19 illustrates two different tokens traveling simultaneously on a 16 Mbs Token-Ring network. IBM has claimed that the use of multiple tokens traveling on the same network can boost efficiency to more than 95 percent on frames larger than 128 bytes.

A Token-Ring Network in Action

A good example of how Token-Ring Networks can be used in a campus environment is IBM's own facility at Research Triangle Park in North Carolina. As Figure 20 illustrates,

Figure 19.
Multiple Frames Traveling on a 16 Mbs Token-Ring Network

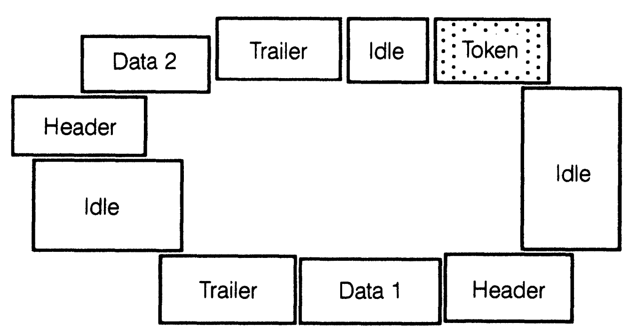
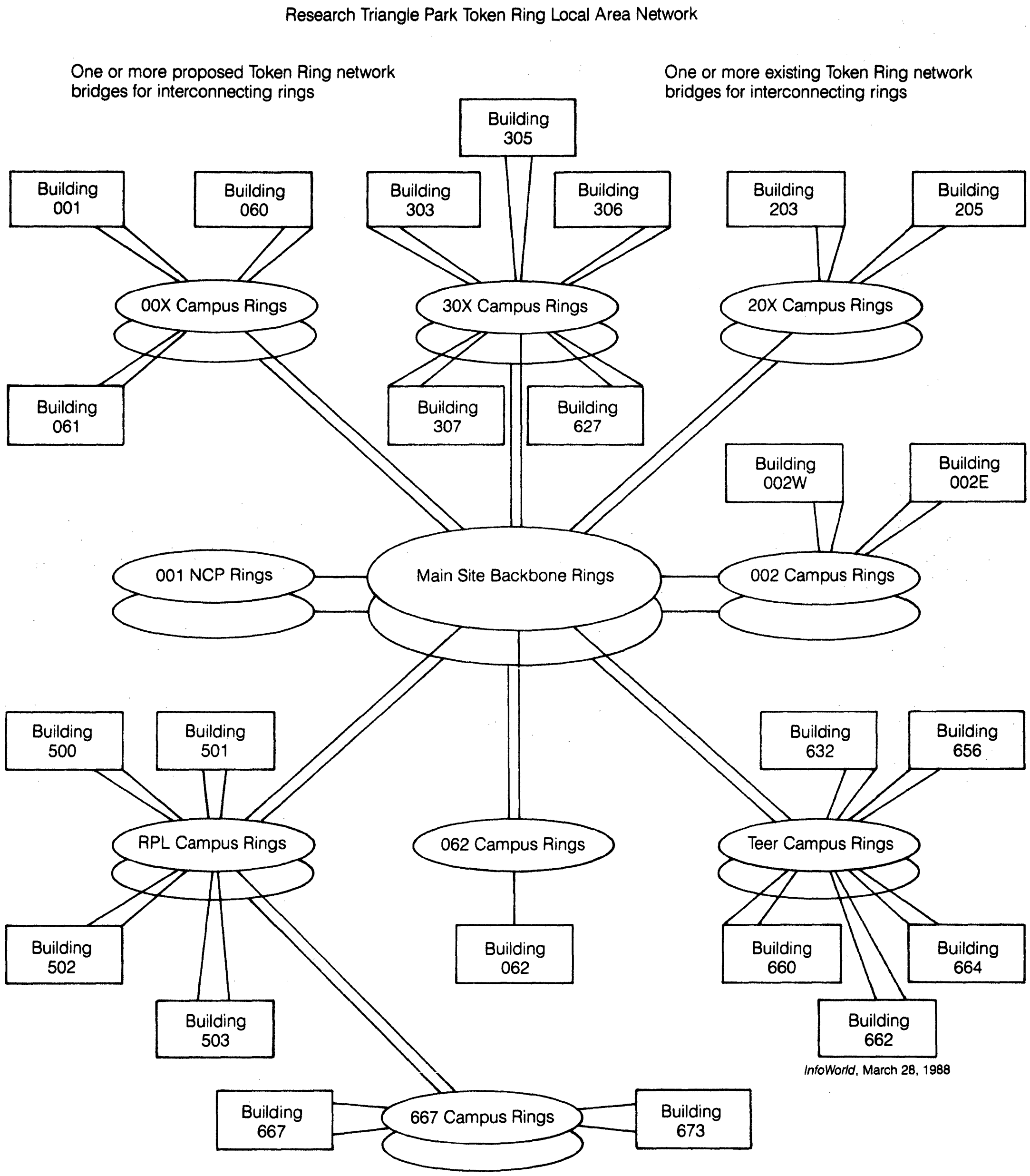


Figure 20.
IBM's Token-Ring Network at Research Triangle Park



IBM has chosen to use backup Token Rings for all LANs with more than 200 users. This redundancy feature illustrates how a company must balance the expense associated with redundancy with the cost of a network being out of service. In IBM's situation, the company decided that 200 users or more down at any one time was painful enough to warrant the cost of redundancy. There are at least 39 production mainframe computers and 15 test mainframe computers at Research Triangle that are linked to the Token-Ring Network.

This Research Triangle Park facility features IBM Cabling System Type 1 and fiber cabling for interoffice communications while connections between wiring closets are IBM Cabling System Type 2.

Fiber Distributed Data Interface (FDDI)

As you have seen when examining IBM's Token-Ring Network and its ability to link large multiple ring networks, IEEE 802.5 Token-Ring networks are designed for heavy data traffic yet the speed of these networks (4 Mbs or 16 Mbs) falls well below what a network with heavy traffic really requires. Even using optic fiber cabling which is capable of much faster transmission does not help; the IEEE 802.5 standards were not designed to take advantage of optic fiber's incredible transmission speed.

Developed by the American National Standards Institute (ANSI) committee X3T.9, the *Fiber Distributed Data*

OSI layer	FDDI layer
Data link	MAC
Physical	PHX (physical layer)
	PMD (physical medium dependent)

Figure 21.
The Structure of FDDI

Interface (FDDI) standard is a counter rotating token ring capable of covering a very large area (200 km) while transmitting data at 100 megabit/second speed, a standard that ensures compatibility with IEEE 802.5 token-ring networks by maintaining the same frame fields found in that standard.

The FDDI model is reasonably consistent with OSI model. As Figure 21 reveals, the FDDI Physical layer is broken down into a Physical Layer Protocol (PHY) that concerns itself with the actual encoding schemes for data and a Physical Medium Dependent layer (PMD) that provides the actual optical specifications. A Media Access Control layer handles token passing protocols as well as packet formation and addressing. Notice also that a set of Station Management standards provides information on such tasks as removal and insertion of workstations, fault isolation and recovery, and collection of network statistical information. SMT utilizes the Connection Management protocol in conjunction with PHY line states to determine whether or not nodes entering the ring are linked together; unfortunately SMT has not yet completed the lengthy process of being formally approved. The effect of this lack of a standard on systems integrators is discussed later in this report.

FDDI was used initially primarily for "back-end" applications such as connecting mainframe systems and mass storage devices and for the backbone network function of connecting different networks together. Today it is ready to take its place along with token bus and token ring as a viable standard for large networks while still performing the backbone function for Ethernet or Token-Ring networks.

What makes FDDI so appealing despite its expense are its speed of transmission and its dual ring approach that offers built-in protection against system failure. One major difference between IEEE 802.5 Token Ring networks and an FDDI network is that a Token Ring network circulates one token at a time on a 4 Mbs network and perhaps 2-3 on a 16 Mbs network. On the 4 Mbs network version, a sending station transmits its token and then waits until the token is returned to it by the receiving station with an acknowledgement that the message has been received before passing the token to the next workstation on the ring. In an FDDI network the workstation sending a message passes on the token immediately after transmitting the message frame and *before* receiving an acknowledgment that the message has been received. The result of this procedure is that several message frames can be circulating around the ring at any given time. Another difference between FDDI and Token Ring which enhances network speed is the use of a restricted token; it is possible to keep other workstations off the network while a time critical task is performed. A Timed Token Protocol also used in IEEE 802.4 networks ensures that low-priority messages will not clog up a network during peak hours. Timed Token Protocol uses both synchronous and asynchronous transmission. Workstations utilize a certain amount of transmission bandwidth defined for use as synchronous service while the remaining bandwidth is used by workstations that transmit signals asynchronously when the token service arrives earlier than expected. They continue to do so until the expected time of token arrival when they switch to synchronous transmission.

In addition to greater speed of transmission, FDDI networks enjoy a built-in redundancy that protects against system failure. The FDDI standard specifies a dual ring, one primary ring to carry information while a secondary ring carries control signals. Figure 22 shows a typical FDDI network. Notice that should a break in cabling result between stations A and B it would be possible for them to

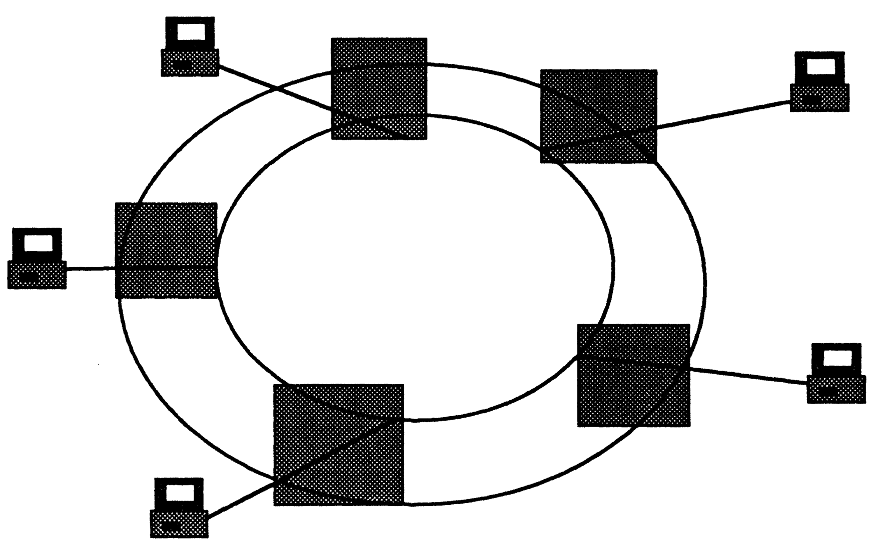
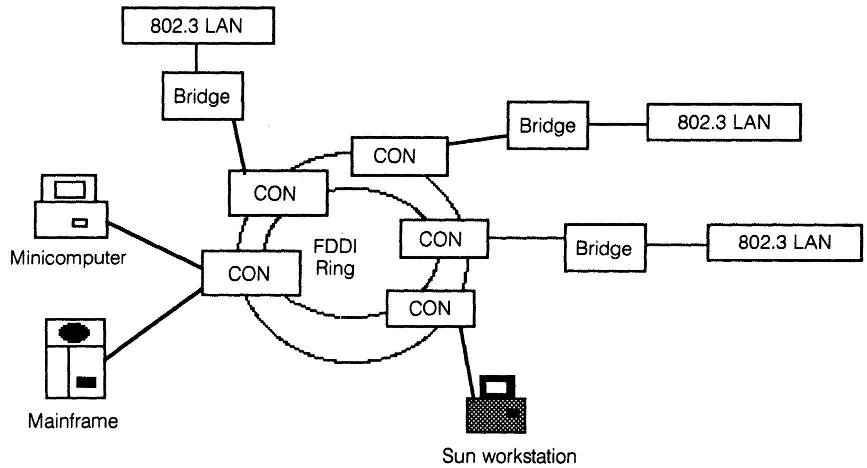


Figure 22.
A Typical FDDI Network Using Dual Cabling

Figure 23.
Concentrators on an FDDI Network



continue to communicate through station C which is acting as a wiring concentrator. Note that it is possible to send data over both sets of cabling traveling in opposite directions so that if there is not a break in the cabling a transmission speed of 200 megabits/second is possible.

The Basic Components of an FDDI Network

The FDDI standard defines the network components required including a *Single Attachment Station* (SAS), a *Dual Attachment Station* (DAS), and wiring concentrators. Single attachment stations are attached to wiring concentrators using a star topology as illustrated in Figure 23. Notice that the concentrators can include mainframes, minicomputers, and high-performance workstations. A cable break with a single attached workstation will not bring down the network because the concentrator is able to bypass this workstation and continue transmitting and receiving information.

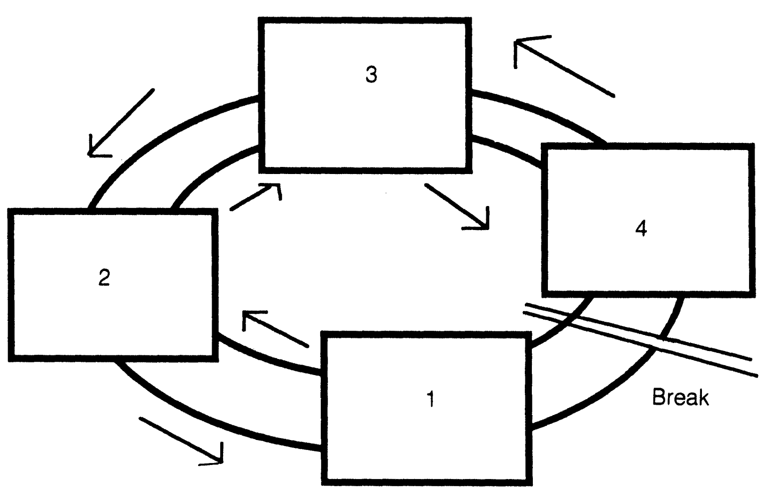
These wiring concentrators can be very attractive for a systems integrator because they can connect anywhere from 4-16 workstations to the network at a much lower cost than using dual attached interfaces. Also, devices attached to concentrators can be switched off without affecting the network. Devices attached with a dual attachment interface can have an adverse effect on an FDDI network if they are placed out of service because the FDDI network might assume the device is defective and try to remedy the situation by wrapping around itself, a phenomenon I discuss in more detail very shortly. Many industry experts expect FDDI network designs to utilize concentrators for PCs and other workstations while using the more expensive but system fault tolerant dual attached interfaces for minicomputer and mainframe links.

Double attached workstations on an FDDI network use dual cabling. The dual attached interface provides system fault tolerance through its redundancy. In Figure 24, you see an FDDI double attached network with a token traveling in one direction. A break in the cabling causes the network to perform a function called *wrapping* in which it activates the second ring to bypass and isolate the failed station. The network will continue to operate but performance will decrease. Some vendors offer an optical bypass cable on their double attached interfaces so that the connection between the right side of the network and left side of the network can be maintained even with a cable break.

FDDI permits a maximum of 1,000 connections with a maximum fiber-optic path length of 200 Km. The cabling specified in the FDDI standard is 62.5/125 micron multimode optic fiber with light generated from long-wavelength LEDs transmitting at 1300 nm. Each station on an FDDI network functions as an active repeater, which helps explain why FDDI networks can be so large without signal degradation.

As I touched upon earlier in this report, another reason why FDDI networks are so fast is their use of a restricted

Figure 24.
Wrapping After a Cable Break on an FDDI Network



Preamble	Starting delimiter	Frame control	Destination address	Source address	Data	Frame check status	Ending delimiter	Frame status

Figure 25.
The FDDI Frame Format

token mode. Normally FDDI networks use synchronous transmission to transmit large blocks of information. Restricted token mode reserves all asynchronous bandwidth for a dialogue between two workstations that wish to use this type of transmission. All other workstations continue to broadcast in synchronous mode while this dialogue is taking place.

The FDDI Frame

The FDDI frame illustrated in Figure 25 can have a maximum size of 4,500 bytes, which makes it ideal for large file transfers. The Preamble field synchronizes the frame with each station's clock using a signal that consists of 16 1-bits. Unlike an 802.5 network where the Monitor Station sets its master clock and then all other stations use its transmission signal to set their own clocks, FDDI uses a distributed clocking approach where each station's NIC sets its own clock. The Starting Delimiter signifies the beginning of the frame and is followed by a Frame Control field. This field indicates whether the transmission is synchronous or asynchronous, what size address will be used (16-bit or 48-bit), and the type of frame to be found (an LLC frame or MAC control frame).

The Destination and Source Address fields are self-explanatory. If the first bit in the Destination field is set, then we have a group address indicating a broadcast message to all workstations on the ring. The Data field is followed by a Frame Check Sequence (FCS) field that uses a 32-bit CRC to check for errors. The Ending Delimiter (ED) field indicates the end of the frame except for the Frame Status (FS) field which is used for a station to indicate it has detected an error.

Using Multiple Tokens on an FDDI Network

Figure 26 illustrates the FDDI token. A station wishing to transmit seizes a token, absorbs it, and then adds its information to form a frame. This station begins transmitting and continues to do so until it runs out of information to send or its token-holding timer expires.

The frames transmitted are repeated by other workstations around the ring. The Destination workstation identifies its address on the frame and copies the information, checks for errors, and then sets a bit in the FS field if it detects an error before transmitting the frame. When the frame is returned to Source workstation, it retransmits it if there was an error detected or it purges the frame.

The moment a workstation releases a token, other workstations can seize it and begin transmitting frames. Workstations do not need to wait for the return of the frame they sent before releasing the token.

Integrating FDDI Networks With Existing LANs

Primarily because the Station Management (SMT) layer of FDDI has not been approved yet by the IEEE, the burden of integrating FDDI networks with existing LANs falls on the systems integrator. This link requires some kind of router or bridge, and at the moment there are vendors offering completely different types of solutions.

The major approaches to linking FDDI nets with existing LANs are data encapsulating bridges, translating bridges, spanning-tree bridges, and source-routing bridges. The data encapsulation method used by companies such as Fibronics, Inc., bundles data into FDDI format using proprietary algorithms. It takes a packet from a LAN and encapsulates it within an FDDI packet for its trip around an FDDI ring. The encapsulating bridge at the receiving end strips the encapsulation from the packet and sends it along its way. The encapsulating and de-encapsulating processes utilize proprietary algorithms that make this approach vendor-specific and incompatible with other vendors' encapsulating bridges as well as other types of bridges.

The translating bridges approach offered by companies such as FiberCom, Inc., readdress data using a protocol-independent method that is open rather than proprietary. A translating bridge will take a packet from a LAN such as Ethernet and translate the packet into FDDI packet protocol. At the destination end, a second bridge will translate the FDDI protocol back into the original LAN protocol.

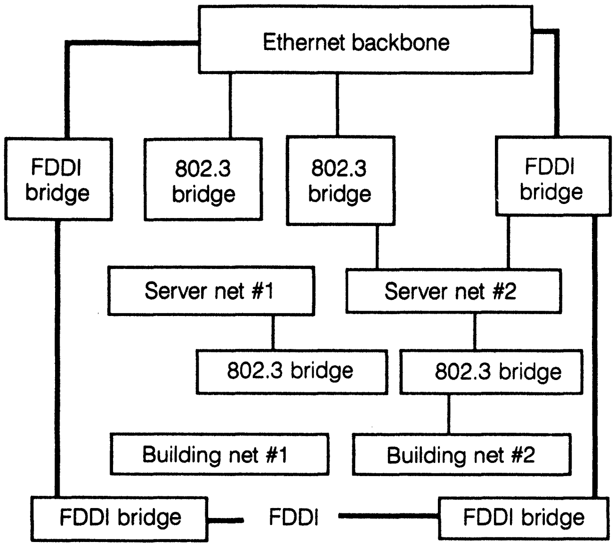
The other two types of FDDI bridges also use the translating bridge approach. The Spanning Tree is utilized by Ethernet LANs and is defined by the IEEE 802.1 standard. The Spanning Tree approach is known as a transparent approach because the bridge is transparent to a workstation when it transmits a frame with a destination address and no routing information. The bridges read these destination addresses and copy those frames that they will pass across to the other side. The Source Routing approach that is used by 802.5 networks has long been advocated by IBM. It requires workstations to maintain tables of network addresses. The workstation must plan out the route it wants its frame to travel. Recently IBM backed a proposal by both the 802.1 and 802.5 groups to the 802.1D committee that will enable Source Routing and Spanning Trees to interoperate. It probably will be at least another year before you see this new standard formally adopted. One problem faced by FDDI developers is that whatever approach they use to bridge FDDI to existing networks, they must be able to handle the multiple protocols found on larger networks. While FDDI does support a number of protocols including TCP/IP and IPX, the FDDI devices having framing formats must be fine-tuned to handle each protocol.

As a systems integrator needing solutions now, you can do something when designing FDDI networks to ensure interoperability in the future. Some vendors are willing to guarantee that they will replace their current FDDI bridges that use proprietary methods with SMT compatible equipment when the standard is formally approved. You might

Preamble	Starting delimiter	Frame control	Ending delimiter

Figure 26.
The FDDI Token

Figure 27.
Microsoft's Use of an FDDI Backbone



InfoWorld, June 4, 1990

want to look for this guarantee as a way of ensuring that you will not lock your client into a proprietary path that might be a dead end.

XTP and FDDI

With IBM's backing, a new high speed LAN protocol called eXpress Transfer Protocol (XTP) has been formally proposed to ANSI. Corresponding to the network and transport layers of the OSI model, this protocol is intended for use with FDDI and higher speed networks to overcome throughput limitations of software-based protocols.

Presently, high-speed networks like FDDI only realize a fraction of their potential speed. FDDI using TCP/IP might only achieve speeds of 15 Mbs because of the amount of time consuming buffering required. XTP will provide low data transmission delay and high throughput with minimal overhead.

FDDI-II Might Be Better Than Rambo II

Most movie sequels disappoint, but a new version of FDDI might provide solutions for network designers who want to use FDDI in an environment where voice, data, and video are integrated. It divides the available bandwidth into a maximum of 16 separate and equal full-duplex channels using a time-division multiplexing approach.

FDDI-II will help systems integrators link intelligent PBXs as well as process control facilities. FDDI-II stations will be compatible with FDDI stations, but will run at the slower speeds in mixed environments. FDDI-II is not yet finalized, but it is coming soon perhaps to one of your neighborhood networks.

An Example of FDDI in Action

Imagine having to design a way to link several networks spread out over a 250-acre campus. That's the problem that faced Microsoft at its Redmond corporate headquarters. Microsoft has developed a corporate policy on how it cables its networks and subnetworks. Its buildings are all prewired for Ethernet subnetworks.

Microsoft's E-mail demands exceed 1.6 million messages each month, this coupled with all the large files programmers handle routinely creates quite a load on any network backbone. Microsoft originally had linked its various networks using standard LAN bridges. When it established its FDDI backbone, it kept its original LAN bridges for system fault tolerance.

Figure 27 illustrates the role of Microsoft's FDDI backbone. To give you some idea of the level of traffic on this network, Microsoft must link more than 9,000 PCs and more than 3,000 Macintosh computers. The PCs are linked to the backbone through Ethernet cards and an Ethernet-to-FDDI bridge. The Macintosh computers are linked to the backbone asynchronously via a LocalTalk connection to Farallon Starcontrollers, which are then connected to Ethernet using Kinetics Fastpath gateways. The Ethernet network includes the Ethernet-to-FDDI bridge already mentioned. Microsoft might replace some of these cards with Ethernet cards in the Macintosh computers.

Summarization of Token-Ring and FDDI Concepts

The 802.5 standard describes a local area network using a token ring, noncontention approach. Currently both 4 Mbs and 16 Mbs versions are available, but a systems integrator must be careful in network design because of cabling distance restrictions. The 802.5 standard is broad and general enough that many vendors have experienced interoperability problems. The Open Token Foundation is an attempt at cooperation among 802.5 vendors.

The Fiber Distributed Data Interface (FDDI) is a 100 Mbs transmission speed fiber-optic network modeled on the 802.5 standard that does have some significant differences. It uses a completely different error correcting scheme that features a double ring and a wrapping technique. Network timing is handled by each workstation rather than by 802.5's Monitor station. Rather than the token reservation system used by 802.5 networks, FDDI uses a timed-token system. Finally, FDDI is strictly a fiber network and does not permit the use of twisted pair or coaxial cabling. ■

LAN Design Concepts: AppleTalk and Arcnet

In this report:

Datapro Summary

Although both AppleTalk and Arcnet LANs have been enthusiastically accepted by networking customers, neither technology has reached standard status by the IEEE standards group. AppleTalk and Arcnet LANs differ from the majority of popular LANs in many ways. In order to be used in a corporate-wide computing environment, network planners must be aware of the unique attributes of these networking systems.

The Macintosh Environment

The Macintosh operating system contains many advanced features not available to PC users until recently with OS/2. Macintosh advocates will argue that this operating system is still far superior. It can address up to 16 megabytes of RAM, and with System 7.0 it will provide true 32-bit addressing rather than the current industry's 24-bit addressing.

Systems integrators are excited about System software 7.0. This version of the Macintosh operating system offers a number of features that can enhance network communications. Virtual memory can mean up to 128Mb of memory on some Macintosh models. This operating system will use virtual memory to swap out the most recently used part of its memory for storage on the hard disk. The memory manager will swap it back for other material when this information is requested.

Another connectivity feature associated with System 7.0 is its *Interapplication Communications Architecture* (IAC). This structure permits one application to send data or commands to other applications located on the same machine or on a different machine on the network.

IAC consists of three key components. A program-to-program communication module inside System 7.0 will handle routing as well as store-and-forward functions from one task to another. An editions manager automatically updates information changed in one application in other applications. IAC also includes AppleEvents, a module that enables developers to have one program send commands to another program. Finally, System 7.0 includes *Data Access Manager* (DAM). Applications communicate with DAM, which in turn communicates with appropriate databases. Apple believes that developers will want to write drivers for DAM that will facilitate internetwork database queries. An application program could use DAM to query databases on VAX, IBM, and Apple computer networks. As long as the database programs had DAM drivers, this process would take place and be transparent to end users.

Still another connectivity feature provided by System 7.0 is distributed file sharing software known as Macintosh *FileShare.* This Apple Filing Protocol (AFP) compatible software enables users to make all or part of a hard disk available to other networked users. What Apple did, in effect, was to build into System 7.0 some of its own AppleShare file server software so that every workstation has the ability to serve as a file server for other workstations.

This Datapro report is a reprint of Chapter 6 "The Nonstandard Apple Networks and Arcnet Standards," from pp. 133-154, book #3577 *Linking LANs: A Micro Manager's Guide*, by Stan Schatt. Copyright © 1991 by Windcrest Books, an imprint of TAB Books, a division of McGraw-Hill, Blue Ridge Summit, PA 17294 (1-800-233-1128 or 1-717-794-2191). Reprinted with permission.

The Building Blocks of AppleTalk Networks

Because the terminology used in the Macintosh world is so different from that found in the PC world, it is useful to begin our look at Macintosh networks by defining the basic building blocks. An AppleTalk local area network includes a Macintosh workstation, a hardware network interface and cabling, appropriate protocol software, and a network operating system.

While IBM PCs and compatibles equipped with AppleTalk network interface cards can operate on an AppleTalk network, I focus initially on an all Macintosh network.

The Macintosh Workstation

The Macintosh comes with a built-in hardware interface to the AppleTalk network. If you want to use Macintosh computers on an Ethernet or Token Ring LAN, you will have to add the appropriate network interface card. This AppleTalk interface is also found on Apple's LaserWriter printers.

LocalTalk

Here's where the terminology might get a bit confusing. The Macintosh's hardware interface contains the low-level software responsible for transmission and media access control to Apple's *LocalTalk* cabling system, which uses shielded twisted pair wire. In other words, the Macintosh's LocalTalk interface is capable of packaging bits into packets and then transmitting them following the network bus's rules for media access at a speed of 230.4 Kbs.

Cabling

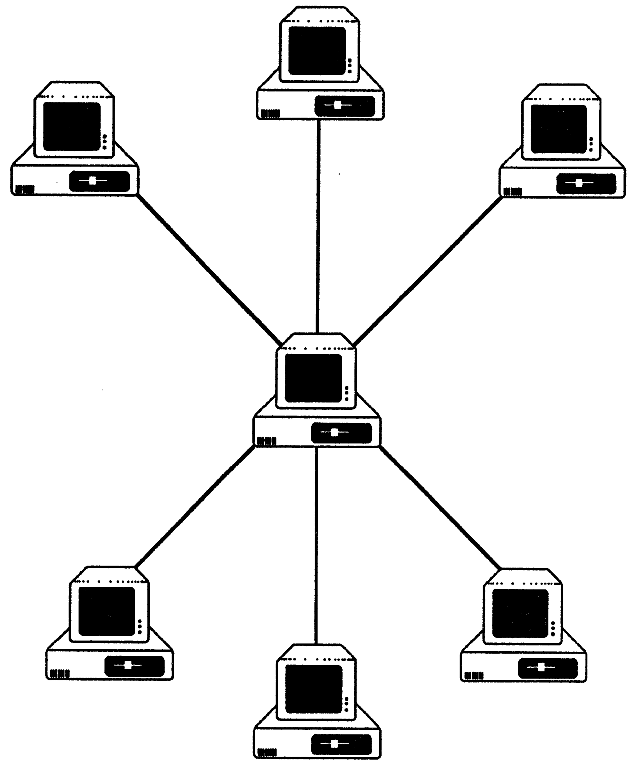
Apple's shielded twisted pair cabling uses RS-422 signaling for transmission and reception over LocalTalk and requires repeaters for distances greater than 1,000 feet. Two repeaters can extend the network to a maximum of 3,000 feet. One limitation of the LocalTalk shielded twisted pair cabling is that its bandwidth is limited, and likely to become saturated if a network grows much beyond 25 users. One solution is to use cabling with greater bandwidth such as the coaxial cabling associated with Ethernet. This report looks at Macintosh Ethernet networks later. A second solution is to utilize AppleTalk's ability to create a series of subnetworks and bridge them together. This report looks at creating AppleTalk bridges later when you examine AppleTalk's protocols.

PhoneNet

Many systems integrators select Farallon Computing's PhoneNet cabling scheme for their AppleTalk networks because of its ability to use existing telephone cabling which consists of unshielded twisted pair wire. What is equally attractive to the systems integrator is PhoneNet's ability to be configured a number of different ways. A *passive star* topology as illustrated in Figure 1 consists of wires branching out of a central block. Telephones normally are wired in this manner with each office receiving its own branch. This passive star arrangement only requires a single pair of unused wires. They have an effective length of up to 4,000 feet and can support up to six branches.

An *active star* topology utilizes a device known as a star controller. This intelligent device has 12 ports that can support 12 physical networks, each up to 3,000 feet of cable. This star topology facilitates network management.

Figure 1.
A PhoneNet Passive Star Topology



Farallon offers software that provides error detection, configuration control, and other network management functions. Figure 2 illustrates this active star topology.

A third topology offered by PhoneNet is the daisy chain. Using this approach, each workstation is daisy chained to another workstation as illustrated in Figure 3. This type of architecture is ideal for a very small network, perhaps an office with just a few workstations and a printer. This topology has a 2,000 foot maximum length.

A backbone topology consists of PhoneNet telephone wall wiring serving as a data highway. Workstations and other devices connect to the backbone through wall boxes. This topology can extend up to 4,000 feet and is designed for heavy traffic. While standard PhoneNet runs at the AppleTalk network speed of 230.4 Kbs, an Ethernet version is available that runs at 10 Mbs over telephone wire.

CSMA/CA

LocalTalk uses a multidrop bus, the same scheme used by Ethernet. It uses a media access method that is similar to Ethernet's CSMA/CD but slightly different. Carrier Sense Multiple Access With Collision Avoidance (CSMA/CA) is an approach in which a workstation that wants to transmit information first senses any activity on the network. If a collision occurs, all workstations avoid additional possible collisions by waiting for the network to be idle a specified amount of time plus an additional amount of random time that varies from station to station.

Unlike CSMA/CD where hardware detects a collision, CSMA/CA works on the premise that a collision *might* have occurred. It relies on "handshakes" between sending and receiving workstations. When a workstation does not receive an appropriate handshake or control packet reply

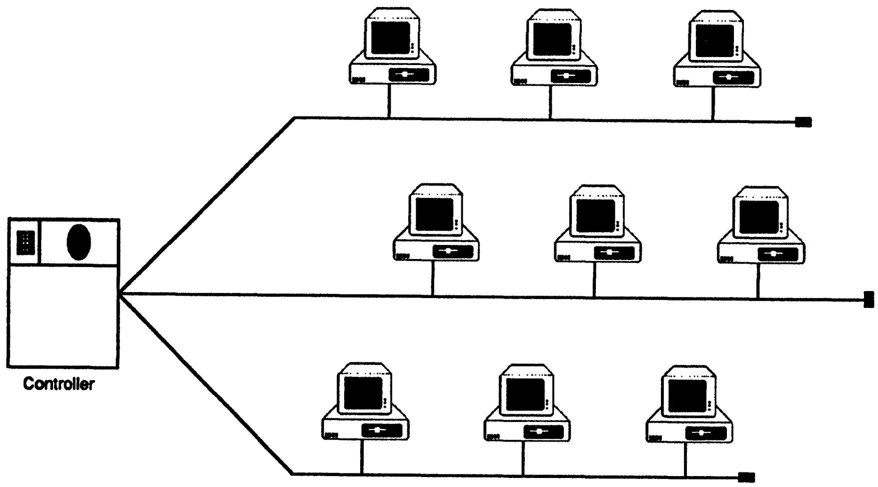


Figure 2.
PhoneNet's Active Star Topology

to its request to send information, it infers that a collision has occurred and begins the process of requesting the right to transmit all over again. To examine this process in more detail, look specifically at the LocalTalk protocol responsible for this data link access method.

LocalTalk's Link Access Protocol (LLAP)

LocalTalk has its own link access protocol known as the LocalTalk Link Access Protocol (LLAP). Figure 4 illustrates the LLAP packet. A Preamble indicating the start of a frame is followed by Destination and Source Node Identifiers. These 8-bit addresses can range from 1 to 127 for user nodes and 128 to 254 for server nodes under AppleTalk first version which is known as Phase 1. The number 255 is reserved for a broadcast Node ID. When a workstation starts up, it randomly assigns itself an ID and then tests to see if another node already is using it by sending out an inquiry control packet.

The LLAP type field indicates whether a packet is carrying control information or actual data. The data length field indicates the amount of data the packet carries; an LLAP packet can carry between 2 and 600 bytes of data. A Frame Check Sequence field is used for error checking and is followed by a trailer flag (01111110) field and an Abort field that indicate the end of a frame. LLAP uses a technique known as *bit stuffing* to ensure that no other field contains more than five consecutive 1-bits. It will insert a 0-bit after five consecutive 1-bits to ensure the Flag field's uniqueness. A receiving workstation's LLAP will reinsert the appropriate 1-bit so that the data can be read correctly.

How Workstations Transmit Using LLAP

When a workstation using LLAP wants to transmit, it checks the network until it has been idle for at least 400 microseconds. It then waits an additional random period. The source workstation then sends a request-to-send packet to the destination workstation, which replies with a clear-to-send packet. The source workstation then transmits its data packet. If a source node does not receive a clear-to-send control packet, it assumes that there has been a collision and once again waits until the network has been idle for at least 400 microseconds.

Broadcast transmissions work somewhat differently. A source workstation waits at least 400 microseconds and then an additional period of time before sending out a request-to-send control packet with a broadcast address. It then checks the network for a maximum period of 200 microseconds before broadcasting its transmission. The source workstation will attempt up to 32 retransmissions before reporting failure.

AppleTalk

AppleTalk refers to the entire suite of protocols that comprise Apple's own layered network architecture.

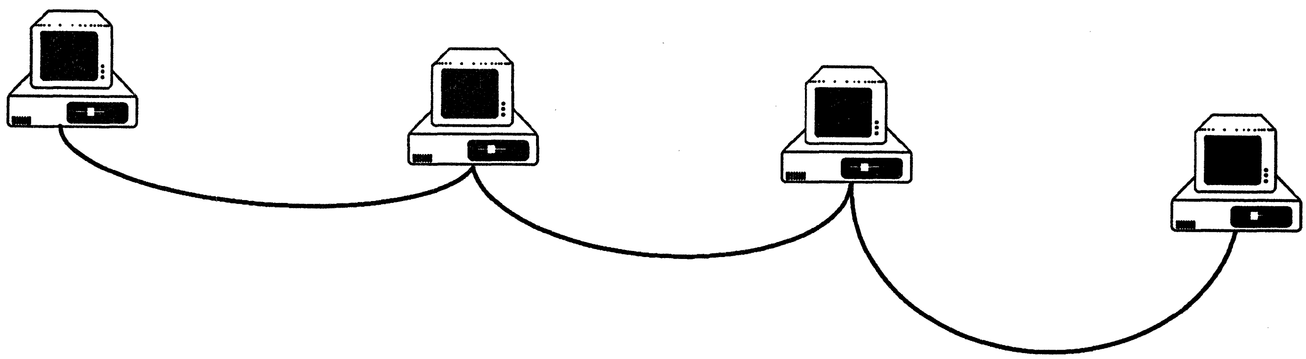
AppleTalk Phase 1

AppleTalk Phase 1 imposed the limitation under LocalTalk of no more than 32 nodes in a single network including workstations, servers, and peripherals including the LaserWriter. It was possible, however, for several AppleTalk networks, each one limited to a maximum of 32 nodes, to be bridged together.

Apple uses the term *internet* to define two or more local area networks linked together with a router or gateway. When a router combines network segments into an internet (Figure 5), the networks remain independent of each other. The networks connected together by routers are known under Apple's terminology as *zones.* A zone consists of logical grouping of networks on an internet; it is very important to realize that a zone need not be identical with the original LAN's physical configuration. In other words, nodes that share functionality (accounting, word processing, etc.) might be grouped together in a zone by a network manager even though they are located on different LANs on the same physical site comprising the internet. Often a backbone network (also seen in Figure 5) is used to reduce traffic congestion between networks on an internet. The name of the game is to reduce the number of routers necessary to transmit information from one node to another node.

In addition to its own LocalTalk, Apple's AppleTalk Phase 1 provided drivers for Ethernet (EtherTalk). This meant that it was possible to design a Macintosh network using coaxial cabling and Ethernet NICs to achieve 10 Mbs transmission speed. The 8-bit field AppleTalk assigned for node addressing meant that even under Ethernet there was a limitation of 254 network nodes which included workstations, printers, modems, etc. For some corporations, this restriction simply was not acceptable.

Figure 3.
A Daisy Chain Topology



AppleTalk Phase 2

AppleTalk Phase 2 added improved internet routing for up to 1,024 interconnected AppleTalk networks. It provided extended resource grouping to support up to 256 zones per network. Even more appealing to the systems integrator concerned with connectivity, it provided drivers for Token Ring (TokenTalk) and enhanced drivers for Ethernet.

Perhaps the most significant change under Phase 2 was replacing Phase 1's 8-bit address field that limited AppleTalk workstations to addressing a maximum of 254 nodes with a 24-bit field that provided the ability theoretically to address more than 16 million different nodes (2^{24}). One serious problem with this change was that Phase 1 packets are invisible to Phase 2 nodes because of the incompatibility of the addressing schemes. While Phase 2 can handle the addresses of newer Apple LaserWriters including the SC, NT, and NTX, it cannot handle the addresses of the older LaserWriter and LaserWriter Plus models. Also, some software (particularly network management software offered by third-party vendors) might work fine on Phase 1 networks but fail under Phase 2.

Moving to Phase 2 meant that companies wishing to use Ethernet needed to upgrade the EtherTalk drivers on each Macintosh node. Much more serious, however, Phase 2 required upgrading all hardware and software network routers. Routing is completely different under Phase 2. A network manager has limited options. One choice (Apple's suggestion) is to bring down the entire AppleTalk network and upgrade all nodes to Phase 2. This expensive option might help your Apple stock rise, but it might cause your company stock to decline rapidly. A second option is to divide the AppleTalk network into several small internets and then selectively upgrade each one independently. This option will disrupt internet traffic, but careful design of the internets could minimize this problem. The third option is the equivalent of a dual protocol stack. Routers could be outfitted with both sets of drivers. This approach will work, but it also will cause serious degradation in network efficiency because of the overhead imposed.

AppleTalk Protocols and the OSI Model

As I mentioned earlier in this report, AppleTalk is a network architecture consisting of a suite of layered protocols. Figure 6 illustrates how it compares to the OSI model.

The Presentation Layer

In the Presentation layer, the AppleTalk Filing Protocol (AFP) provides the basis for the entire network's file structure; it also provides a key network interface for file server software. AFP provides any translation required for native AppleTalk file system calls to be understood by a file server.

The PostScript protocol is also found at the Presentation layer; it ensures that the network can communicate with PostScript printers.

The Session Layer

AppleTalk protocols found in this layer are concerned primarily with the Session layer's responsibility for establishing a communications session. The *Session Protocol* handles the correct sequencing of datagrams when they arrive out of order. It also takes responsibility for ensuring that datagrams are the correct size and that there are break points during conversation sessions. Break points are used to re-establish disrupted sessions without having to start over again from the very beginning.

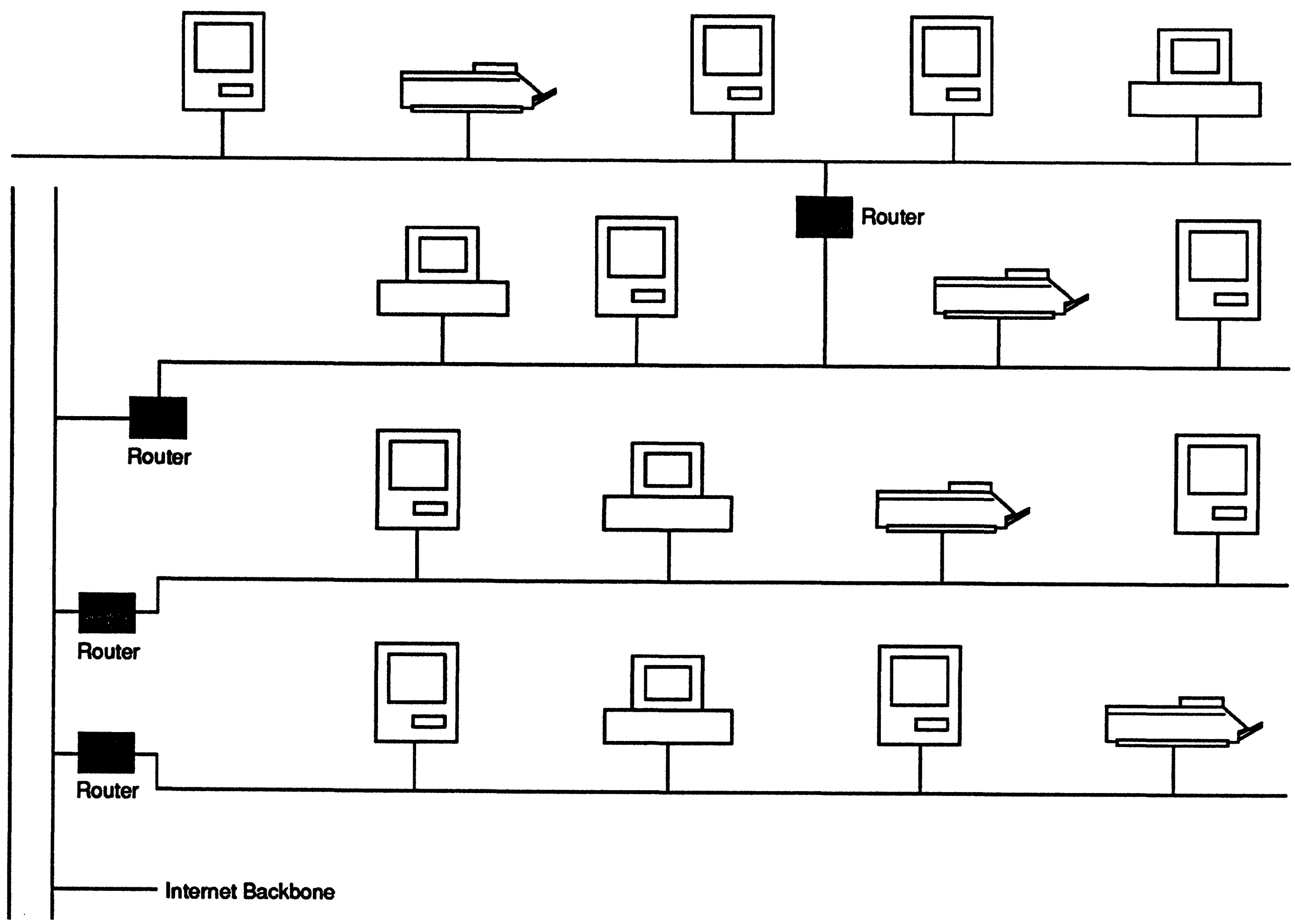
The *Data Stream Protocol* establishes the actual communications session. It can provide full duplex service, detect and eliminate duplicate datagrams, and request retransmission when needed to ensure error-free service.

The *Printer Access Protocol* handles streaming tape systems as well as streaming printer sessions for devices that use this approach. Finally, the *Zone Information Protocol*

Figure 4.
The LLAP Frame

Preamble	Destination Node ID	Source Node ID	Type	Length	Data	FCS	Flag	Abort

Figure 5.
An AppleTalk Internet



maps a network into a series of zone names. This information is used by both bridges and routers when they determine their delivery path for a particular message.

The Transport Layer

The Transport layer is primarily responsible for planning the routing of a datagram from one network to another. The actual details of this routing are handled by the Network layer. The Transport layer must determine the types of transport services required. What level of error checking is necessary? Should acknowledgement of delivery be required? These are the types of questions that this layer must address.

The *Routing Table Maintenance Protocol* (RTMP) keeps a table that indicates the number of bridges that must be crossed (the number of "hops") to transport a datagram from one network to another network. This protocol also determines alternate routes should the primary route fail.

The *Name Binding Protocol* matches workstation server names with internet addresses. Users need not concern themselves with the appropriate internet address because this protocol makes the process transparent. The *AppleTalk Transaction Protocol* is an extremely critical component of OSI compatible Transport layers because it provides the acknowledgement of a datagram's error-free delivery that some network applications require.

Finally, the *Echo Protocol* enables the destination workstation to echo the contents of a datagram to the source network workstation. This service lets the network know that a workstation is functioning and also provides a measure of the round trip delays encountered on the network.

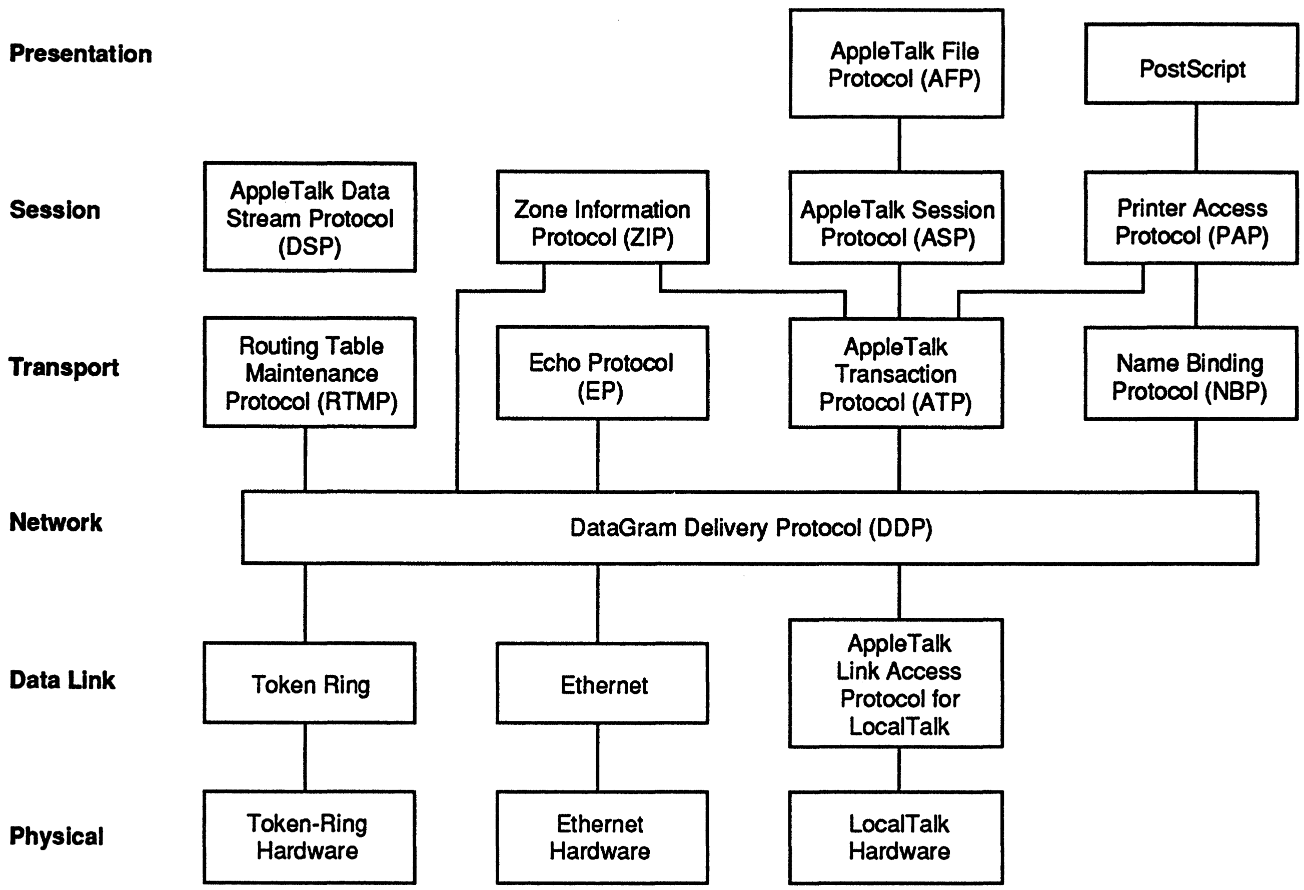
The Network Layer

As we pointed out when examining the Transport layer's functions, the Network layer performs the nitty-gritty details required for the transport of datagrams from one network to another. The *Datagram Delivery Protocol* is able to address specific logical ports on different networks and to establish the route a datagram will take. This protocol is able to use the Transport layer's Name Binding Protocol to translate a server's name into an internet address and then use the Routing Table Maintenance Protocol to establish a network path for the datagram.

The Data Link Layer

As I pointed out in my earlier discussion of AppleTalk Phase 2, AppleTalk has link access protocols for Token Ring (TokenTalk) and Ethernet (EtherTalk) as well as its own LocalTalk hardware. The drivers AppleTalk provide are able to handle the different addressing schemes, different sized packets, and different media access methods required by these different types of networks. Figure 7 illustrates how an EtherTalk packet differs from the LocalTalk

Figure 6.
AppleTalk Protocol and the OSI Model



packet discussed earlier in this report. Notice it contains the Ethernet-specific fields required for communications on an Ethernet network.

Assume that a user sitting at a Macintosh workstation decides to send a file over an Ethernet network with AppleTalk protocols. The File Manager in the Macintosh operating system (System) requests the file from another module within the Macintosh operating system, the SCSI Manager. The SCSI Manager issues the commands required for the hard disk to locate the file and copy the data into RAM.

The Macintosh's central processor takes this file and packages it into AppleTalk packets, which are sent to the EtherTalk Link Access Protocol (LAP). The Macintosh NuBus Manager (also part of the Macintosh operating system) sends these EtherTalk packets over the NuBus to the Macintosh's Ethernet NIC. This circuit card translates the EtherTalk packets into electrical signals and uses a CSMA/CD media access method to transmit these Ethernet signals at 10 Mbs.

The Physical Layer

The Physical layer on an AppleTalk network defines the hardware required to communicate with Token Ring, Ethernet, and LocalTalk networks. AppleTalk Phase 2 supports a Token Ring card called the TokenTalk NB (for NuBus). This 32-bit Macintosh II network adapter has its own 68000 processor and memory as well as its own support software. This software enables the Macintosh to use all the higher level AppleTalk software including AppleTalk Filing Protocol on a Token Ring network. This support software also has an SMB (Server Message Block) file transfer program that enables Macintosh workstations on a Token Ring network to exchange files with PCs on the network. The program transfers a PC file into the Macintosh, but does not support sharing the PC file while it is on the PC. AppleTalk cannot let a Macintosh application run on an SMB server because this server does not have the intelligence to understand AppleTalk file structure.

One limitation at the present time that might be corrected soon is that despite its current hefty price ($1,250), TokenTalk NB only supports Token Ring's 4 Mbs transmission speed and not the 16 Mbs standard.

AppleShare

AppleShare is Apple's own network file server software built on a foundation of AppleTalk Filing Protocol (AFP). It is popular with corporations that like to have a one vendor solution for their networks, but it does have some serious drawbacks for the systems integrator.

If you have been an Apple dealer or have purchased Apple products, it should come as no surprise that AppleShare is expensive—over four times as expensive as NetWare. Because Apple does not provide such key network services as electronic mail and print spooling as part of AppleShare, they must be purchased from Apple (if available) or third-party vendors. The Macintosh must serve as

Figure 7.
An EtherTalk Packet

Ethernet Destination	Ethernet Source	Type	AppleTalk Destination	AppleTalk Source	AppleTalk Type	Length	Pad

the file server on an AppleShare network, while PCs can participate as long as they run the PC AppleShare software that translates Sever Message Blocks to AppleTalk Filing Protocol calls. Figure 8 illustrates AppleShare PC in action. Apple offers Ethernet NICs for the Macintosh II, but it does not offer these boards for all Macintosh models. One advantage of AppleShare for schools is that it will run on Apple IIe and IIGS computers.

AppleShare is adequate for relatively small networks that do not require elaborate security or services. Under AppleShare, for example, there are no folder-level passwords.

Other Major Macintosh Network Options

Several other major network options for the Macintosh are covered in the following sections.

Transcendental Operating System (TOPS)

Sun's Sitka division offers TOPS (Transcendental Operating System) software. TOPS uses existing LocalTalk cabling as well as its own TOPS Filing Protocol (TFP) protocols to create a peer-to-peer network. Rather than using a centralized file server, each workstation on a TOPS network runs TOPS software and becomes a file server to other workstations on the network. Systems integrators have been attracted to TOPS because of its ability to link Sun workstations, which use Sun Computer Corporation's own *Network File System* (NSF) protocol with networked Macintosh computers; the connection is easy because Sitka TFP software for the Macintosh also contains the NSF protocol.

NSF is a service that provides transparent access to remote file systems on a network. As far as network workstations are concerned, data located anywhere on the network appears to be located on just another local hard disk drive.

Another reason TOPS networks have become so popular for systems integrators is the ease with which IBM PCs and compatibles can be integrated with Macintoshes. The company was one of the first to offer an AppleTalk NIC for PCs along with a DOS version of TOPS software. PCs on a TOPS network *publish* the files they want to share with other network users who *mount* the files they wish to access. TOPS provides a built-in conversion utility to handle IBM and Macintosh formats of popular programs such as Microsoft Word and Lotus 1-2-3.

A high-performance version of the TOPS LocalTalk NIC is known as the TOPS FlashCard. One advantage of this adapter is that it can transmit network information between PCs at 770 Kbs, three times faster than the Macintosh workstations can transmit on the same network. The Macintosh units require a FlashBox LocalTalk accelerator to run at this speed. A relatively new TOPS NIC provides FlashCard performance for microchannel architecture PS/2 models, something that many large corporations and major Macintosh sites such as General Electric have been requesting. Other TOPS features that systems integrators might find attractive include versions for Microsoft Windows and OS/2, support for TokenTalk, and gateways from its electronic mail program (Inbox) to Unix Mail and IBM's PROFS mail system.

One serious limitation of TOPS' distributed approach to file serving is that if you are using workstation B's files and that computer crashes, you could lose whatever changes you had made to your file since the last time you saved it. Also, if someone is accessing one of your files and you turn off your computer without issuing a warning, you could corrupt your version of the open file such that your system will crash the next time you try to access it.

Com's 3 + Macintosh Network Software

3Com was one of the first vendors to link Macintosh workstations and PCs using Ethernet. Its 3 + Macintosh software has a number of features that make it very appealing for a systems integrator. The 3 + software and AppleShare

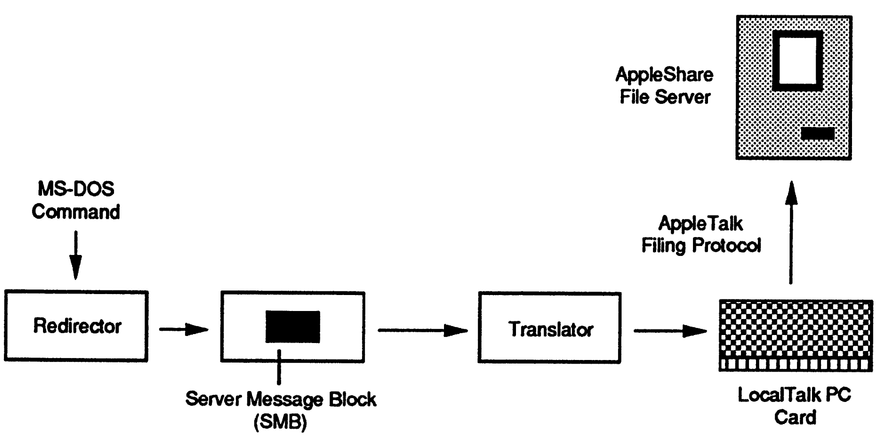
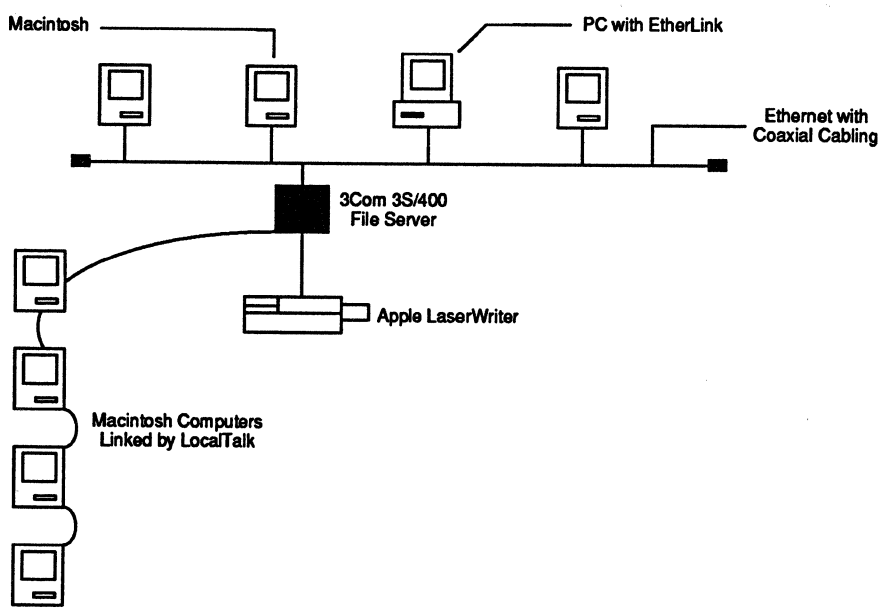


Figure 8.
AppleShare PC in Action

Figure 9.
A 3 + Network



are compatible and can coexist on the same network transparently to users and to applications that use the AppleTalk Filing Protocol (AFP). As Figure 9 illustrates, 3Com's sophisticated file servers, NICs and operating system software make it possible to link Macintosh workstations on a LocalTalk network with both Token Ring and Ethernet networks.

Because 3Com began operations as a pure Ethernet company, it is not surprising that XNS is its upper level protocol. 3Com packets are routed in the Transport layer using XNS's Sequenced Packet Protocol (SPP). You might remember that Novell took SPP and modified it to create NetWare's SPX packet format which, unfortunately, is just enough different to be incompatible with the 3Com format. 3 + software for the Macintosh also includes AppleTalk's Datagram Delivery Service which is used by AppleShare file servers to support AppleTalk bridges. Because 3 + software also includes IBM's Server Message Block (SMB) protocol, DOS and OS/2 workstations are able to share the same file simultaneously with Macintosh workstations because the network operating system understands both these two very different methods of requesting files from a file server. When a Macintosh workstation under the 3 + Macintosh network operating system issues a record lock on data in a file on the 3Com server, the 3 +

Figure 10.
3 + Macintosh Software Translates File Manager Requests

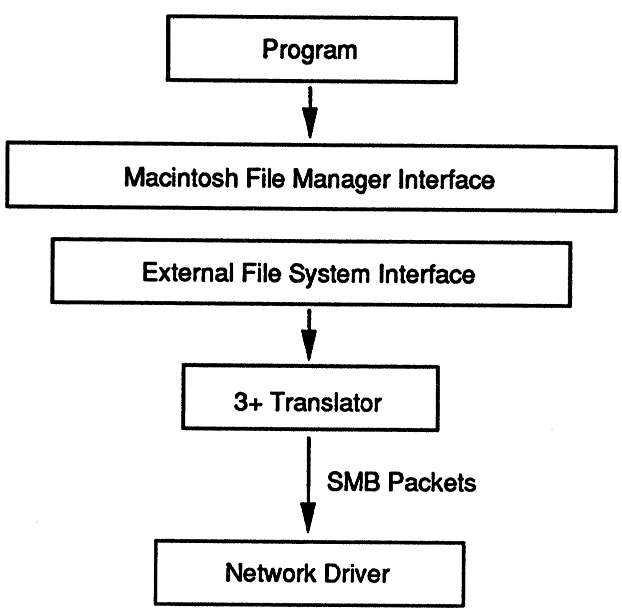
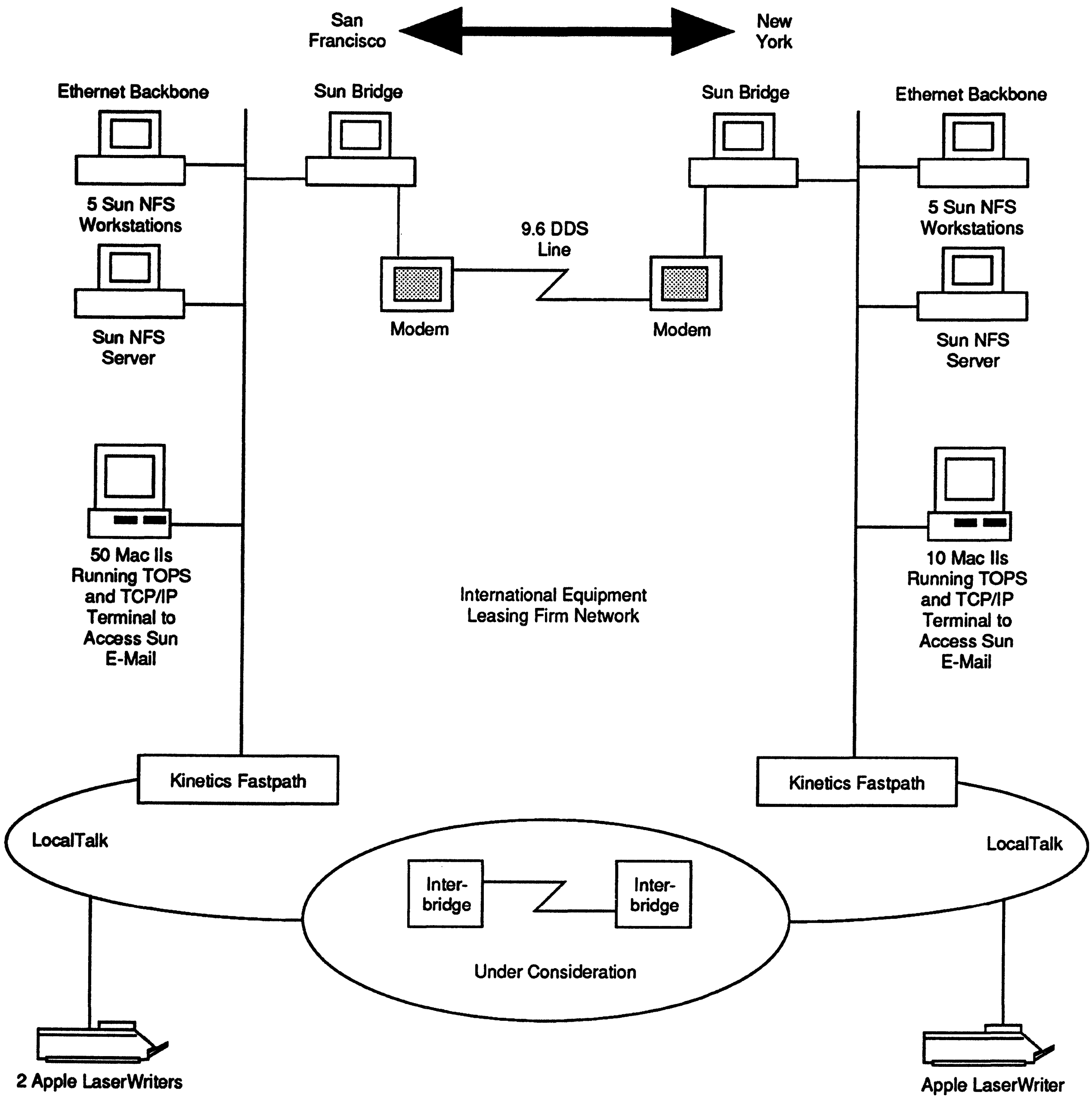


Figure 11.
Protocols on a 3Com 3 + Open Network and the OSI Model

Application	XNS Name Server Microsoft Redirector
Presentation Session	AppleTalk File Protocol XNS Courier
Transport Network	XNS MINDS AppleTalk Datagram Delivery Services
Data Link Physical	MNP Token Ring Ethernet

Figure 12.
Macintosh Computers Networked and Linked Across the Country



software intercepts the Macintosh File Manager call to perform this task and translates it into the corresponding DOS operating system call. The file server never knows or cares whether the command is coming from a Macintosh or from a PC. Figure 10 reveals how 3 + software translates these File Manager requests.

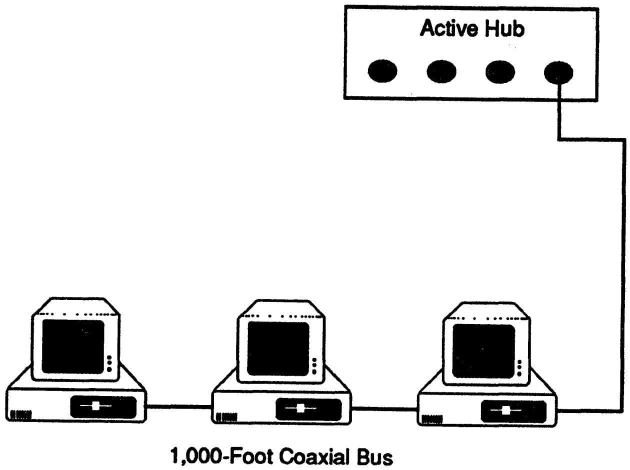
Figure 11 reveals that the 3 + software also includes Microcom Network Protocol (MNP) at its Data Link level; this protocol has become a de facto communications standard and explains in part why 3 + software is so strong in the area of internetwork communications.

Because 3 + Mail service operates in both the DOS and Macintosh worlds, it is possible to transmit electronic mail easily and efficiently. This mail service works with the MultiFinder, so it is possible to display mail on a desktop while running other applications. The 3 + Reach/MCI software provides a transparent mail gateway between 3 + Mail and MCI Mail while 3 + Route software provides internet routing service between remote servers.

What attracts many companies with both Macintosh and PCs on a network to 3Com are the connectivity features built into the 3 + software. 3Com offers TCP/IP communications, 3 + NetConnect software to provide routing among Ethernet, Token Ring, and AppleTalk, LAN to WAN connections via an X.25 wide area network.

Systems integrators are also attracted to 3 + software by an efficient tape backup system (3 + Backup) and system fault tolerant disk mirroring or server mirroring (3 + Fault

Figure 13.
An Arcnet Bus Topology with an Active Link



Tolerant). One 3 + requirement that does not please systems integrators is the proprietary 3Com family of servers. While they are optimized for 3 + software, they cannot be used as workstations should a company later move to a different network operating system. They become $20,000 doorstops or paperweights if a company shifts to NetWare.

NetWare for the Macintosh

NetWare for the Macintosh uses the AppleTalk Filing Protocol (AFP) used by Apple's own AppleShare file server. In fact, to a Macintosh workstation, the NetWare file server appears in the Chooser desk accessory as an AppleShare file server. NetWare adds many features not found under AppleShare, though, including enhanced security, better printing facilities, fault tolerance, and a transparent link with the 50-60% of corporate PC networks running NetWare.

A major advantage of NetWare for the Macintosh is that a Macintosh network can use Intel 80286 or Intel 80386 based microcomputers as servers. NetWare also can utilize 2 gigabytes of disk storage on a single file server. The prices of these AT compatibles and corresponding hard disks are far cheaper than what you will find in the Macintosh world. NetWare's print spooling is compatible with the AppleTalk Printer Access Protocol so that PC workstations can access Apple LaserWriter and ImageWriter printers.

NetWare is able to translate the AppleShare file server format to its own NetWare Control Protocol (NCP) that runs on a NetWare file server. A NetWare utility translates Macintosh file names into DOS file names since the latter are limited to 8 characters without spaces while Apple files can contain up to 31 characters.

NetWare's major strength is its security. Directory trustee rights and file attributes can be limited by the network supervisor. Directories can be hidden and users can even be restricted to certain hours on the network. The system fault tolerance includes the ability to duplicate file servers, link UPSs, and perform transaction tracking so that if the network is interrupted, transactions in midstream will not corrupt the files that are opened.

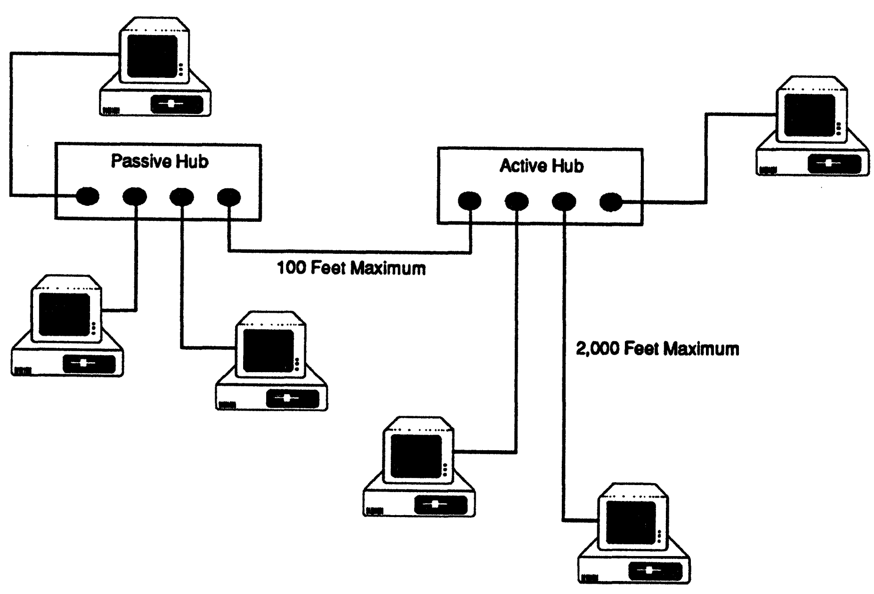
A Corporate AppleTalk Network in Action

In 1988 an international equipment leasing firm decided to move away from its mainframe Prime computer and design a network that would use Sun workstations running the UNIX operating system to run its lease-analysis program. Macintosh computers running TOPS are networked and provide all the office automation applications. The Macintosh network is connected to an Ethernet backbone at both the company's San Francisco and New York locations (see Figure 12).

TOPS contains the NFS (Network File System) protocol that the Sun systems use so it is easy to link the Macintosh networks to the Sun workstations. The network consultant designing this system has a choice. He can use Apple's EtherTalk network cards in the Macintosh or use lower speed Apple LocalTalk cards. A Kinetic's FastPath Ethernet-to-AppleTalk gateway provides a link to the LaserWriter printers which are linked together using LocalTalk.

A total of 50 Macintosh IIs in San Francisco and 10 Macintosh IIs in New York use a Sun workstation on the San Francisco LAN as a network file server. The TOPS terminal software enables the Macintosh workstations to

Figure 14.
An Arcnet Star with Active and Passive Hubs



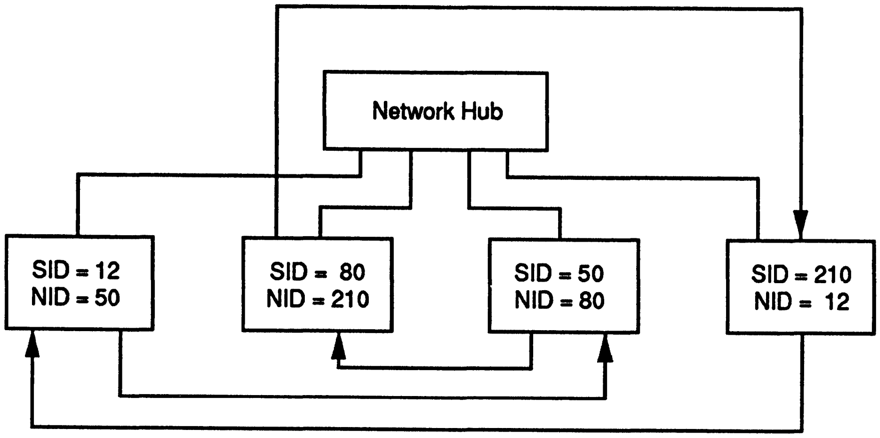


Figure 15.
Arcnet's Token Bus Approach

emulate a Sun workstation and run files on the Sun server as well as use E-mail that connects the Sun and Macintosh worlds.

Arcnet LANs

Attached Resource Computer Network (Arcnet) is one of the oldest LAN technologies dating back to 1977 when it was developed by Datapoint in 1977; since then major vendors that license and support Arcnet include Standard Microsystems Corporation (SMC), Acer Technologies, Thomas Conrad, and Allen Bradley. Arcnet is found in more than one million active nodes, yet it never seems to garner the respect or attention of the industry media. There never has been an IEEE Arcnet standard. Is it dying? According to Standard Microsystems Corporation, a leading Arcnet vendor, Arcnet comprises 20% of all current PC and compatible LANs and is being installed in one out of every four new LAN installations.

Why Arcnet Is Popular

Arcnet is popular with small as well as large systems integrators because it is very inexpensive; an Arcnet card might cost one-third the price of a token ring NIC. It can use a single untwisted shielded pair cable, which most companies can spare; also, it runs on RG 62/U coaxial cabling, the same cabling many companies have for their 3270 terminals. IBM shops that replace their 3270 with PCs can use the same coaxial cabling to build an Arcnet LAN. Companies that do need fiber-optic cabling can also use this medium with Arcnet.

Besides being one of the least expensive and most reliable LAN technologies, Arcnet is also one of the most flexible networks. Adding and removing nodes, initializing the network when first powered one, and recovering lost tokens are functions all handled by Arcnet hardware. It is a joy to install compared to other LAN hardware including Token Ring and Ethernet and much easier to diagnose problems. As you will observe in a few moments, its topology is probably the most flexible found in the industry. It is very important that you keep in mind that Arcnet is hardware that runs virtually any NetBIOS compatible network operating system. You will find LAN Manager and IBM's PC LAN Program running on Arcnet, but the most popular choice for larger businesses is the Arcnet/NetWare combination.

Topology

Arcnet supports bus, star, and distributed star topologies. Each segment of an Arcnet bus can contain up to eight nodes daisy-chained together and can extend to 1,000 feet. Adding an Active Link to a bus segment extends the segment's range another 1,000 feet point-to-point as revealed in Figure 13.

Arcnet's star topology is popular because it is the easiest arrangement to troubleshoot. Also, the failure of a single workstation does not bring down the entire network because each workstation has its own cabling connection with a hub. Up to eight workstations can be connected to a central hub with a 2,000 foot maximum. Passive hubs, on the other hand, are limited to distances of 100 feet. Active hubs can be connected together to build a network with a maximum distance of four miles. Figure 14 illustrates a typical Arcnet star topology with both an active hub and a passive hub.

I mentioned earlier that Arcnet's topology is extremely flexible. It is easy to connect bus segments with an Arcnet star or distributed star. Two port repeaters can connect the bus segment on one side with a star hub on the other side.

Arcnet's Access Method

Arcnet uses a token bus access method. It is a physical star or bus but a logical ring. It is a non-contention network in which each workstation has a turn transmit based on its NIC address set using an 8-position DIP switch. Each NIC knows its own network address as well as the address of the node to which it will pass the token. The highest-addressed node closes the logical ring by passing the token to the lowest-addressed node.

Figure 15 illustrates how Arcnet's token passing network functions. When node 150 has completed its network token time, it passes the token to node 10 and the process around the logical ring begins all over again. Arcnet's token bus technology was developed in a minicomputer transaction-oriented environment where short network bursts consisting of requests for database information placed a premium on moving relatively small data packets quickly and efficiently. Its low overhead and efficient non-contention token bus transmission at 2.5 Mbs makes it competitive with Ethernet under heavy traffic conditions even though it has substantially less bandwidth.

The Arcnet Packet

The major reason Arcnet is incompatible with the IEEE standards is its addressing. It uses an 8-bit locally administered station address format while the IEEE 802 LANs use

Figure 16.
The Arcnet Packet Format

Alert Burst	SOH	SID	DID	DID	Count	Data	CRC	CRC

Alert Burst	Six 1-Bits Identify a Packet
SOH	Start of Header (1 byte)
SID	Source Workstation ID (1 byte)
DID	Destination Workstation ID (2 bytes)
Data	1-508 Bytes
CRC	Error Checking (2 bytes)

a globally administered 48-bit station address. There are simply too many active Arcnet nodes to change the addressing scheme. Instead, Arcnet vendors have been fighting a political battle to persuade an ANSI committee to approve an Arcnet standard. Figure 16 describes the Arcnet packet format.

The Alert burst consists of six consecutive ones that identify a packet. Notice that the Source and Destination IDs are one byte and two bytes respectively, far different than the IEEE Ethernet and Token Ring formats examined earlier in the report. Notice also how relatively small the data field is; it is dwarfed by Ethernet's approximately 1,500 bytes. When Ethernet has to retransmit its packets because of frequent collisions, the beauty and simplicity of Arcnet becomes apparent.

Arcnet's age is apparent, though, in the way that nodes communicate with each other using a character-oriented protocol. If node 20 wants to send a packet to node 40, it would first send a Free Buffer Enquiry (FBE) which asks if node 40 is ready to receive a transmission. Node 40 responds with an Acknowledgement (ACK) or a Negative Acknowledgement (NAK) if it declines the FBE. When node 40 receives the data packet from node 20, it checks the CRC to ensure that there was no error in transmission. It then transmits an ACK to node 20 indicating that everything arrived successfully. If it fails to send an ACK, node 20 determines that there must have been an error in transmission. Assuming the message did arrive correctly and an ACK was transmitted, node 20 would then issue an FBE to the next node scheduled to use the token.

Arcnet Plus at 20 Mbs

One of the most recent developments has been the announcement of a 20 Mbs Arcnet Plus network. Datapoint has the proprietary rights to this technology but has licensed SMC and NCR to sell products. Arcnet Plus is able to dynamically vary its data signaling rate so that it is backwardly compatible and interoperable with standard Arcnet. Network supervisors need not pull any existing Arcnet NICs or cabling but can introduce Arcnet Plus network segments selectively such as on backbones to optimize network performance.

Arcnet Plus supports packet sizes from 12.5 to 4224 bytes compared to a maximum size of 516 bytes with standard Arcnet. The network supports up to 2,047 nodes per segment over standard coaxial or twisted pair wire compared to standard Arcnet's 255 nodes. This enhanced version of Arcnet will support the IEEE 802.2 48-bit addressing as well as the 802.2 media access control layer. This will enhance compatibility with other 802 networks. Arcnet Plus will also continue to support traditional Arcnet addressing.

Network design when mixing the two Arcnet technologies might be a headache for systems integrators. Standard Arcnet NICs cannot recognize addresses above 255, so it is important to minimize communications between these two sides of a mixed network. Similarly, current hubs only transmit at 2.5 Mbs; new hubs might take a while to reach the market.

Summary

Every Macintosh and Apple laser printer has a LocalTalk interface built into its hardware. While AppleTalk's transmission speed is 230.4 Kbs, AppleTalk's suite of protocols also includes drivers for Ethernet (EtherTalk) and Token Ring Network (TokenTalk). Transcendental Operating System (TOPS) offers peer-to-peer network software and hardware for the Macintosh while both NetWare and 3 + network operating systems offer sophisticated centralized file-server Macintosh networks which incorporate AppleTalk protocols and can handle Apple's own AppleShare file server packets.

Arcnet is a popular nonstandard LAN technology that transmits at 2.5 Mbs with an Arcnet Plus version transmitting at 20 Mbs. Arcnet offers a variety of topologies including the star, bus, and a mix of both. Arcnet's low overhead and small-sized packets traveling under a token bus methodology results in a very efficient yet low-cost network. ■

Local Area Network Connectivity

In this report:

Datapro Summary

As local area networks play an ever-growing role in the infrastructure of an organization, efficient interdepartmental flow of information between local area networks is required. Achieving company-wide integration is a tall order with much confusion over business dynamics, technical issues, and industry standards. Current standards and connectivity strategies need to be examined in order to implement an effective integration system.

Introduction

The need for inter-organizational communication is the key to effectiveness in competitive markets. Information is viewed as a strategic resource and the network facilities that transmit data have become essential to the efficient functioning of the business. In the 1980s local area network (LANs) became the standard for departmental data processing. In the 1990s management and movement of data across LANs will become the key to gaining strategic advantages over competitors and integrating applications across the business.

Achieving integration is no easy task. The ad hoc deployment of LANs in earlier years has made integration difficult and complex. The challenge to network managers in the future will be to find ways to obtain the cost benefits of integrated LANs without disrupting current operations and systems.

This report examines connectivity issues dealing with internetworking multiple local area networks (LANs). Due to the increasing need for interdepartmental transfers of data and the dynamic state of industry standards, this is an area that demands management attention.

This Datapro report is a reprint of "Local Area Network Connectivity" by David C. Brueggen and David (Chi-Chung) Yen, pp. 105-113, from *Computer Standards & Interfaces*, Volume 11, Number 2, 1990. Copyright © 1990 by Elsevier Science Publishers. Reprinted with permission.

To understand the current state of interconnectivity. LAN development will first be examined along with benefits achieved by integrating LANs. Standards will also be examined from the point of promoting and inhibiting LAN interconnectivity. Knowledge of current standardization issues is critical in LAN integration. Both OSI and IEEE efforts in standardization will be examined along with their impact to LAN connection. To gain further insight into the technical aspects of LAN connectivity, transmission media, frame formats, protocols, and network configuration should be considered. Finally, three strategies for LAN connection will be discussed. Relative advantages and disadvantages of each strategy, as well as devices used to facilitate each type of connection are presented in an effort to guide management in successful systems integration decisions.

Background and Benefits

The personal computer has played the greatest role in LAN development. The price-performance advantages of PC's have pushed processing power onto the end user's desk. The decade of the eighties witnessed a dramatic decrease in the price, and an exponential increase in processing power of PC's. This caused the PC to become the standard for departmental office automation. The development of departmental LANs has taken a similar path. As LAN technology progressed and prices decreased, LANs were increasingly employed as a departmental processing solution. This is reflected by an IDC survey which showed that the installed base of LANs increased from just under 100,000 in 1985 to a projected 3.6 million by 1992.[1] Like any high growth industry in its infancy there were a large number of vendors each with unique LAN solutions. As the need to connect networks throughout the company became more critical to staying profitable in competitive environments, network managers have had to deal with incompatibility problems caused by a large installed base of different LAN products.

In an effort to meet dynamic business needs, departments have been forced to use currently available technology to attempt to overcome incompatibility of existing LANs.[1] Solving the incompatibility problem has been part of the problem itself. LAN vendors, in providing network solutions, have contributed to the larger problem of product incompatibility by installing a large base of ad hoc solutions.[2] Indeed, to solve incompatibility problems vendors have focused on a confusing number of devices designed to serve one interconnection need or another.

Meeting the incompatibility challenge has become of paramount importance to network managers because of important benefits achieved by connecting LANs. The first benefit is in the area of systems management. With all organizational LANs connected together, systems management can take place from a centralized location by a network manager trained in network management. Network installation and security would also be more consistent under centralized control, while still allowing true distributed processing.

With networks tied together and servers centralized, data integrity is enhanced because storage-backup can be controlled. This would eliminate inconsistent and missing backup logs which can be a problem at a departmental level.

Another benefit is in the transfer of electronic mail. Large companies may have installed several different brands of electronic mail system on different LANs. Connection devices allow electronic mail to be quickly and transparently transferred between departments running on separate LANs. This increases productivity and decreases internal mail costs. Files can also easily be transferred. File transfer is especially useful in large development shops doing design and programming for large systems. Integration is also an issue in mergers and acquisitions. Often network equipment of the merged company will be incompatible with the equipment of the parent company. In these cases integration tools and technique can be used to overcome incompatibility, saving the company the time and cost of a major network overhaul.

Issues/Considerations Which Affect LAN Connection

Standardization Issues

OSI Effort

To understand connectivity issues dealing with LANs, current standards must be examined. Standards have long been a source of confusion with LANs. The lack of standards when LANs were in their infancy caused the large number of network solutions that are seen today. To promote interconnection between the various architectures emerging in the 1970s the International Standards Organization (ISO) in March of 1977 developed a seven layer reference model for computer networking called the Open Systems Interconnection (OSI) architecture.[3]

Table 1. Lower Three OSI/IEEE Layers

Layer	Function	Attribute
3. Network	Route data from one node to another	Routing software
2. Data link		
2.1 Logical Link Control (LLC)	Provide interface to network layer	Connection-oriented Connectionless
2.2 Media Access Control (MAC)	Provide interface to physical layer	CSMA/CD Token bus Token ring
1. Physical	Transmission of bit streams across a particular medium	Fiber optics Coax Twisted pair

The primary purpose for the OSI model is to provide a basis for coordinating development of standards that relate to the flexible interconnection of systems using data communication facilities.[3] The lower layers have been where the greatest amount of standardization has been achieved. and are the layers that directly influence the interconnection of LANs (Table 1).

The physical layer is responsible for the transmission of bit streams across a particular transmission medium.[3,4] The second layer, the data link layer, is responsible for providing data transmission from one node to another and for shielding higher layers from concerns about the physical transmission medium.[3,4] The network layer is concerned with routing data from one network to another.

Recently the US Government announced that it would purchase only OSI compatible products.[5] It is hoped this will lead to further commitment to the OSI model. Vendors have already started marketing OSI compatible products. OSI still has some way to go in terms of standardization though. Despite rapidly growing acceptance, full-fledged OSI is not here yet. The following concerns about OSI have been voiced:[6]

1. In the near and medium term, there will be areas that remain undefined leaving potential users with the problem of systems with only partial OSI conformance.
2. Even though vendors claim conformance to a standard, they may not interconnect because of inconsistencies in the standard as interpreted.
3. Implementing a layered architecture adds on unacceptable amounts of overhead and could slow processing.
4. Data security is a perceived weakness in "open" communications systems.

IEEE Effort

In the US the Institute of Electrical and Electronic Engineers (IEEE) has been developing standards regarding architectures for LANs. The IEEE standards were developed by the IEEE 802 committee in cooperation with the American National Standards Institute (ANSI), and issued as joint IEEE/ANSI standards in 1985.[3] Project 802 addresses only the lowest two layers of the OSI model, the physical and data link layers.[3,4] In the 802 architecture, the data link layer is divided into two sublayers:

Logical Link Control (LLC)— This sublayer allows the network layer of the OSI model to access the services of the

Table 2. Impact to LAN Interconnection

Organization	ISO	IEEE
Effort	OSI model	Project 802
Concentration	General network architecture	LAN
Layers	Application Presentation Session Transport Network Data Link Physical	Logical Link Control (LLC) Media Access Control (MAC) Physical
Implications to LANs	General standards open to interpretation	Family of standards serving proprietary interests

LAN without regard to how the network is implemented. There are two types of services within the LLC sublayer: connectionless, and connection-oriented.[7] Connectionless service provides for delivery of messages without the need for permanent end-to-end connections in the network layer.[8] In contrast, connection-oriented service establishes end-to-end connection with the network layer for the duration of the session.[8]

Media Access Layer (MAC)— This sublayer is concerned with the access control method that determines how the physical transmission is controlled.

At the MAC level, the 802 committee addressed several standards.[3,4,8,9] IEEE 802.3 is based on Ethernet and makes use of the MAC protocol CSMA/CD running baseband or broadband on coax cable. IEEE 802.4 is based on token-bus and regulated by token passing. At the physical layer 802.4 specifies the use of broadband and carrier band running on coax cable. IEEE 802.5 is based on a token-ring protocol using shielded and unshielded twisted pair.

Impact to LAN Interconnection

Since a large number of architectures were in use when IEEE standards were developed, no single standard could be documented that met the needs of all LAN requirements (Table 2).

The above standards are a compromise of proprietary interests. The development of a family of standards, namely those based on the MAC layer CSMA/CD, token ring, and token bus. has caused confusion among users and increased complexity in interconnecting networks. Although these architectures are compatible, they are based on different protocols. To communicate between these protocols requires conversion to be done at a higher level which increases processing overhead.

LANs have become the accepted choice for departmental distributed processing, although without any standardization on any single LAN vendor, technology. medium, or protocol.[10] This has caused LANs to proliferate without consideration of organization-wide networking. This can be further seen in the fact that at least 75% of Fortune 500 companies have both token ring and Ethernet LANs installed.[10]

Devices Used to Connect LANs

Devices used to connect LANs serve different layers of the OSI, IEEE models. Although advancing technology has begun to blur the distinction between devices, they can generally be broken down into five different classes:

- Repeaters
- Bridges
- Routers
- Brouters
- Gateways

Repeaters: Repeaters work at strictly the physical layer of the OSI model. Repeaters are used not to connect dissimilar networks, but to connect individual network segments to form a larger extended network.[3,11]

Bridges: A bridge is used to connect physically distinct networks.[11] A bridge functions at the second or data link layer of the OSI model. Bridges require that the networks have consistent addressing schemes and packet frame sizes.[9] A Bridge is typically a station that belongs to two or more networks simultaneously.[3,11] The bridge receives all messages on each network, checks the destination address, and sends the messages along if needed.[3,11] Messages that are destined for nodes on the same network are kept local, while messages intended for nodes on another network are passed on. In this way a bridge performs a filtering function. keeping local traffic local, while forwarding only those messages that need to be forwarded.

Routers: Routers interconnect networks at layer 3, the network layer, of the OSI model. Routers allow interconnection of two or more physically distinct networks using different protocols.[9] If a LAN is running TCP/IP protocol, for example, then the router must be designed for TCP/IP. Routers examine the destination of each packet and choose the most reasonable path for that packet.[11] This choice decreases the amount of network data traffic.[11] Routing is useful when a network has many paths between LANs or when LANs are connected through a wide area network (WAN).

Brouters: Vendors have introduced devices that provide some functions of both bridges and routers, termed brouters. Brouters operate between bridges and routers in the OSI model. Like routers, brouters can manage multiple lines and route data packets.[11] At the same time, like bridges, brouters remain independent of the higher protocols of the LANs that they connect. Therefore these devices try to provide the "best" of bothworlds.

Gateways: A gateway is used to interconnect devices of completely different architecture. Gateways provide translation for all seven layers of the OSI model.[9] Gateways handle any conversions that are necessary to go from one set of protocols to another including message format conversion, address translation. and protocol conversion.[12] Since these conversions require a great deal of processing, gateways usually have the lowest performance, and greatest cost of interconnection.[12]

Factors Affecting Connectivity

Considerations that must be taken into account when interconnecting LANs must be examined from the perspective of the seven layers of the OSI model, as vendor solutions aim at different layers of the model. At the physical layer, transmission media and throughput are concerns. At the data link layer, or media access layer, frame formats should be understood. Protocols at the network layer and higher should also be evaluated.

Transmission Media

There are several types of transmission media used by different vendors. Included here are twisted pair, coax, and fiber optics. The two types of signaling methods used on these cables are broadband and baseband.[13]

The two main concerns when considering the type of cable to install, or evaluating existing cabling, are cost and raw transmission speed attainable. Although the controller determines the speed of packets moving through the network, the type of media can effect throughput. Cost increases as one moves from twisted pair, to coax, to fiber optics. Transmission speeds can range from 1 million bits per second (Mbps) on StarLANs using unshielded twisted pair, to 4Mbps on token rings using shielded twisted pair, to 1 to 10 Mbps on Ethernets using several types of 75 ohm coax.[13] Fiber optic cable has only recently been proved usable on LANs. Speeds of up to 100 Mbps have been demonstrated on fiber optic cable.[14]

When making cabling decisions involved with integrating existing networks or adding new networks, throughput should also be considered. If high throughput is required, fiber optics or coax is the best choice. Fiber optics provide a large bandwidth for the greatest throughput, and is relatively inexpensive for short haul usage.[2,13] Simple multimode low-end fiber extenders for local bridges provide LAN-to-LAN connectivity under one mile and operate at 850 nanometers.[3] Fiber can carry a cost advantage in some cases because it eliminates the need for repeaters to amplify signals every 500 meters over larger Ethernet systems.[2]

Coax has traditionally been the most popular medium for transmission. Like fiber, better grades of coax can be used for broadband transmission although the bandwidth is not nearly as large as with fiber. Twisted pair has also undergone changes, and can now be run at speeds of 10 Mbps. In the future, transmission media will probably undergo further improvements, thereby making media choice less important in terms of throughput.

Problems arise when connecting two or more networks operating at different transmission speeds. For example, if a StarLAN network, which operates at 1 Mbps, is connected to an Ethernet network running at 10 Mbps, traffic may enter a connection at a higher rate and be transferred at a lower rate, causing the network to become congested.[15] If packets from the faster network arrive faster than they can be transferred, the bridges buffer will fill and overflow, causing packets to be discarded. This will require the packets to be retransmitted, further slowing network throughput.

When network links become severely congested, delays rise exponentially and time-outs can occur while waiting for packets to traverse a link.[16] In any commercial interactive environment, where a high level of network performance is required, users will be severely frustrated by delays. Bridges employ buffers of usually 64K bytes to 128K bytes to store incoming packets for each side of the bridge.[13] If the appropriate mix of speeds is not possible, the lower speed network will have to be partitioned with additional bridges to decrease congestion.

Frame Formats

At the data link layer of the OSI model, frame formats needs to be considered. When the networks to be connected are the same at the MAC level (Ethernet CSMA/CD to Ethernet CSMA/CD) there are no problems in transfer, and full transparency is maintained. But when trying to connect two networks with different formats (Ethernet to Token Ring), extra processing is needed. Such a bridge is transparent only if it is fast enough to preserve full LAN performance across the connection it provides.[10]

To make a token ring frame compatible with an Ethernet CSMA/CD LAN would require reformatting the frame, reversing the bit ordering, and providing a way to recalculate the checksum information.[10] Also, speed buffering must take place. When transferring from an Ethernet LAN to a token ring, even more processing is required since additional formatting is required to account for the token.[10] Since the maximum size of an Ethernet packet is 1500 bytes. compared to 500 bytes for token ring, packets must be broken down into multiple packets and speed buffering must take place.[10]

Another consideration is the type of routing algorithm used, IBM uses the source routing algorithm for bridging token ring LANs, while Ethernet products employ the tree spanning algorithm. Routing algorithms determine how the packet is sent to its destination. The IEEE 802 subcommittee has recently adopted the tree spanning algorithm as a standard for linking all 802 LANs.[10] Network managers should bear this in mind when considering using existing token ring bridging equipment. Resolving incompatibilities between these algorithms and all possible topologies is extremely difficult.[10]

Protocols

A major consideration that must be taken into account when integrating LANs is the protocol suite used. A protocol handles communication across the LAN. Protocols initiate and terminate a session, exchange messages. and recover from errors.[7] Standards at the network level and above have not emerged resulting in a diversity of routing protocols.

Among the current protocols running on LANs are:

NETBIOS— This is an interface supported by IBM LANs. NETBIOS routing protocols are not designed for general transmission across interconnected networks. NETBIOS will support interconnected token ring LANs using the underlying LLC source routing facilities.[3,4]

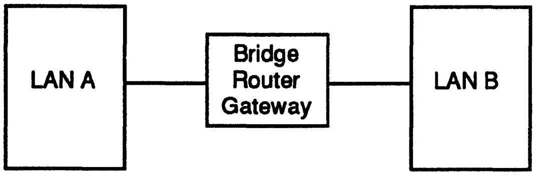
XNS— Xerox Network Systems provides communication between different Xerox products. including the interconnection of Ethernet LANs with each other.[4]

TCP/IP— Transmission Control Protocol/Internet Protocol is a four-layer communication architecture providing network features such as end-to-end communication, packet sequencing, internetwork routing and specialized communication functions.[4]

Network Configuration

When deciding what configuration to use when tying LANs together, size and functionality will usually be the determining factors. In a simple scheme where two LANs

Figure 1.
Point-to-Point Connection



are to be connected together, a point-to-point configuration will be used (Figure 1).

This horizontal design may be extended to more than two networks by adding additional links. When the number of networks to be connected is small this design will provide adequate performance. As the number of LANs to be connected grows, and as more data is transferred between LANs, a horizontal configuration is no longer effective. Performance will decline as data gets backed up by traversing networks. In these cases the configuration will have to be changed by adding redundant links between networks, or a backbone network configuration may be used to centralize the interconnection of LANs.

A backbone is a network that is used to link together several networks (Figure 2). Backbones are primarily used by large organizations to connect together a large number of networks over a long distance, such as a campus or a large building. Backbones offer several advantages to horizontal configurations. If one link goes down, the other networks may continue to communicate with each other. Adding additional networks is also simplified since they need only be connected to the backbone. Network management is also enhanced since reconfiguration will less likely be required.[17]

Because high throughput is needed for a backbone, fiber optics are often used to take advantage of their high bandwidth and ability to transmit over long distances. Because of this, the cost of installation and connection will be higher.

Three Strategies for LAN Connection

There are several reasons a network manager might want to interconnect various LANs. Among them are the following:

Network Integration

The major concern of network managers today is network integration.[18] This is due to the wide variety of ad hoc LAN solutions installed in the past. By integrating these networks the aforementioned benefits of systems management, file transfer, and increase communication can be achieved. Integration needs to take place between LANs of similar topology and access methods (Ethernet CSMA/CD to Ethernet CSMA/CD), and between LANs of dissimilar topology and access methods (Ethernet CSMA/CD to IBM Token Passing Ring). This can be achieved by directly connecting the LANs together or tying LANs together via a WAN backbone. As discussed earlier, connecting LANs of dissimilar topology and access methods presents problems in integration due to incompatible protocols.

Network Expansion

Several problems may arise as a network grows that warrant network expansion. As network grows performance may decrease due to propagational delay or signal attenuation and distortion.[18] A network may also be constrained in terms of distance or the number of nodes connected to the network.[15] For example, IEEE 802.3 Ethernet networks have a limit of 3 kilometers and 1,024 nodes. Here, a second network would be created by splitting the existing network into parts and linking them together. This keeps network congestion down while allowing the number of nodes to increase above the maximum physical constraints in terms of numbers and distance. Traffic capability is also increased due to the fact that local traffic can exist on both networks at the same time.[18]

Partitioning

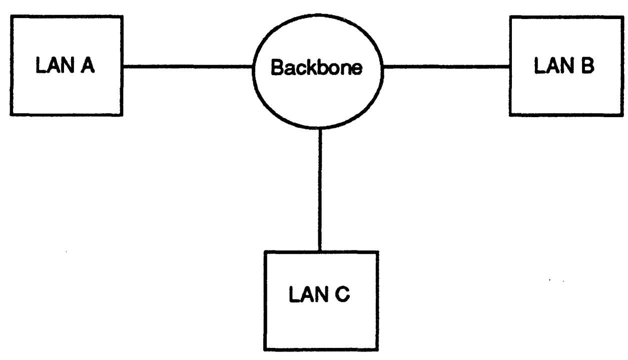
Bridging is also used to partition large LANs into subnetworks). This is primarily done to reduce traffic within the network. Here, two or more nodes might have a high volume of communication between them, causing network congestion.[15] Subnetworking eliminates congestion by partitioning the high volume nodes into their own network, so that most of the traffic between those nodes is localized. There are other advantages in subnetworking. By subnetworking, a manager may use the most cost-efficient media in a particular local area subnetwork without compromising internetworking performance.[19] These networks are also more reliable and easier to maintain as faulty components are easier to track down. Also, network partitioning allows fewer operators to share the workload of a network efficiently.

Implementation of LAN Connection

Once the type of strategy needed has been identified, the next step is to choose the device to achieve interconnection. The manager will find that several of the five classes of devices mentioned earlier will solve connection problems.

When integrating networks all five classes of devices could be used. If the number of LANs being connected is small and they use identical protocols and media then a repeater or bridge can be used. If media differs, a bridge should be used. Using a bridge is the optimum choice because traffic remains localized on each network, increasing overall network performance. As the number of networks grows or the protocols of the networks to be connected change, brouters and routers would be employed to not only transfer messages, but decide the paths the messages take. When the number of LANs to be connected is large, a backbone configuration, as described earlier, should be considered. When integrating networks whose protocols differ on every level a gateway should be used.

Figure 2.
Backbone Configuration



When expanding existing LANs a repeater or bridge should be used. The choice of which one to use will be determined by cost, media used, and whether the traffic will be localized. A repeater boosts the signal strength and sends all messages along to the extended part of the network. A bridge would allow two way communication between the networks, while acting as a filtering device, keeping local traffic local, while allowing packets destined for the other network to pass through. Also using a bridge allows different types of media to be used on each half of the network. If coax is currently being used as a medium, the extended half of the network could employ twisted pair.

If it is decided to partition the network to increase network performance, a MAC layer bridge should be employed. Here, a bridge would partition the congested segment into its own subnetwork, lessening traffic and increasing throughput. This would allow more cost efficient media to be used on either half of the network if needed.

Suggestions to Facilitate Implementation

In making connection decisions it becomes obvious that certain tradeoffs will have to be made to achieve connectivity, especially when dealing with disparate systems. There is no one good answer to connectivity issues. The tradeoff between connectivity and performance will depend on three factors: protocols. medium, and size.

Protocols

Connecting LANs with different protocols decreases performance. Performance decreases as one moves up the protocol stack from the physical, to the datalink. to the network layer, and above. This is due to increased processing required to translate between protocols at increasingly higher layers. Widespread acceptance of OSI and IEEE 802 standards should alleviate these problems in the long run, as all LANs will be able to be bridged at the MAC level. Connection at this level will allow maximum throughput and transparency, although some additional processing will be required when using different access methods. But this does not solve the problem of older systems that do not conform to OSI/IEEE standards. These systems will continue to be connected with ad hoc solutions resulting in poor performance.

Medium

Different media types will continue to affect performance in the short term. Now that fiber optics standards have been established, fiber should become the medium of choice in the future as availability increases and cost decreases.

Size

As the number of networks that are required to be connected together increases, performance declines. A manager should pay special attention to network configuration as size increases. If a large number of LANs need to be interconnected, a fiber optic backbone should be considered to maximize throughput.

Keeping in mind these tradeoffs, a network manager should determine the type of transfers that will be made between systems in evaluating connectivity solutions. The following questions should be considered:

- Will there be a high volume of traffic between networks?
- What type of traffic will this be? Interactive traffic requires high throughput. Bulk transfers from transaction oriented processing may not require high throughput.
- What types of security precautions are needed?
- Does the solution meet the company's long term needs? Have future technological advances been considered?

The answers to these questions should help in evaluating connectivity needs. A manager can then assess the tradeoff between performance and protocols, medium, and size to tailor a solution to connectivity problems using bridges, routers, brouters and gateways.

Conclusion

There are no simple solutions when dealing with LAN connectivity problems. Inconsistent standards, ad hoc implementation of networks, and a variety of interconnection devices have all contributed to difficulties in LAN communication. Future widespread acceptance of LAN standards should clear up most of these difficulties allowing flexible interconnection of LANs. Until then, connection devices such as repeaters, bridges, routers, brouters, and gateways will be used as connectivity solutions. This report has presented strategies to meet connectivity problems, and described devices a network manager can use to implement these strategies. By taking into consideration the issues involved in LAN connection, and employing these three strategies, flexible and efficient LAN communication can be achieved.

References

[1]W. Gewirtz and P. Stuntebeck, Data networking directions. *AT&T Tech. J.* (Nov/Dec 1988) 7-22.
[2]T. Valovic, LANs: strategic development trends, *Telecommunications* (Oct. 1987) 47-50.
[3]J. Martin, *Local Area Networks* (Prentice Hall, Englewood Cliffs, NJ 1989).
[4]W. Stallings, A manager's guide to protocols for local networking, *Telecommunications* (Sept. 1987) 38-44.
[5]J. Hughes, Novell commits to OSI, *LAN Times* (Nov. 1989) 1.
[6]A. Brown and T. Stratton, OSI: the future prospects. *Comput. Comm.* (Apr. 1988) 80-84.
[7]J. Moulton, OSI—an open door for systems integrators, *Mini-Micro Syst.* (Oct. 1988) 80-89.
[8]A. Weissberger and J. Israel, What the new internetworking standards provide, *Data Comm.* (Feb. 1987) 141-156.
[9]T. Routt, SNA to OSI: IBM building upper-layer gateways, *Data Comm.* (May 1987) 120-130.
[10]E. Hindin, Ethernet-to-Token Ring: the gap begins to narrow, *Data Comm.* (Apr. 1989) 47-50.
[11]M. Van Name and B. Catchings, Bridge one way to take the local out of LANs, *Computerworld* (June 1988) c/18.
[12]S. Lowe, At OSI layer 2, learning curve keeps rising, *Data Comm.* (Feb. 1988) 62-64.
[13]A. Krumrey and J. Kolman, LAN hardware standards, *PC Tech. J.* (Apr. 1988) 54-68.
[14]P. Schnaidt, Interop. *LAN* (Dec. 1989) 42-50.
[15]J. Weinstein, Bridging to a better LAN, *Mini-Micro Syst.* (Feb. 1989) 86-93.
[16]J. Weiss, High speed LAN internetworking, *Networking Management* (Dec. 1989) 66-72.
[17]D. Somers and B. Whyte, Network partitioning: an alternative solution to network management (Feb. 1990) 61-71.
[18]W. Stallings, *Local Network Technology* (IEEE Computer Press Soc., Silver Springs, MD, 1985).
[19]R. Craven, The challenge of enterprise-wide internetworking, *Telecommunications* (Oct. 1988) 31-37. ■

Corporate Connectivity

In this report:

Datapro Summary

After a decade of growth, personal computers and LANs have reached a critical point. Technological progress in networking software, equipment, and network management have enabled managers to control their LANs in sophisticated ways. Enterprise networks present a platform for the control and centralization of crucial company information. They also give users throughout an organization structured access to data.

For as long as computers have been used in business, data-processing managers have dreamed of ways to integrate them into a seamless, company-wide system. Originally, the vision was that such a system would consist of a gigantic mainframe swapping data with other mainframes elsewhere in the company. As it turns out, mainframe enterprise computing died an early death. It was the victim of a hardware technology—personal computing—that outstripped the ability of planners and managers (and even computer manufacturers) to keep up.

Before corporations were prepared to implement mainframe-based enterprise computing, people began abandoning centralized computing in droves. Tired of waiting years for software changes and sick of the cost of supporting huge mainframes with little return, they turned to personal computers. This movement finished any chance for mainframes to control all the computing resources in their companies. Today, however, enterprise computing is back, in the form of company-wide networks.

This Datapro report is a reprint of "Corporate Connections" by Wayne Rash, Jr., pp. 215-223, from *BYTE*, Volume 16, Number 9, September 1991.

The Personal Computer Effect

Although the introduction of personal computers was hailed by individuals, the results were mixed from the corporate viewpoint. Personal computers improved productivity just about everywhere they were installed, but it was not always clear that the improvement was worth the loss of control.

In the early days, some people who were tired of waiting for the software they needed wrote their own programs. Many of these amateurs lacked even elementary training in programming; thus, some of the programs they wrote were terrible. A lot of the programs placed the company at risk because they changed data in ways other than expected or intended.

In addition, individuals started creating their own data and storing it on local hard disks. But most of this data was never backed up, once again placing the company at risk if the local disk should fail. This combination of unreliable programming and nonexistent data security caused managers many sleepless nights. Eventually, they turned to LANs.

To a corporation, a LAN has two attractive features beyond its obvious abilities in communications and file sharing. LANs allow some control over the company's information, because the information is stored at a central site: the file server. And LANs give managers a degree of control over what software is available to whom. As a result of these advantages, networks started springing up all over large companies.

Islands of Connectivity

As departments started to adopt LANs, they did so with their own needs in mind. Rarely were LANs adopted with any sort of strategic plan. As a result, many companies found themselves with lots of departmental LANs that did not communicate with each other.

IBM describes these networks as "islands of connectivity." As useful as they are to the departments that have them, these islands are of limited value to the company as a whole. They protect the department's information, but they don't do much for organizing or protecting information on a company-wide basis.

This need for company-wide connectivity created enterprise networking, which has in turn helped to fuel the growth in LAN installations. Organizations of almost every size have found that they can benefit from the ability to share information across the company. They communicate more effectively, plan more consistently (because the information is more consistent), and worry less about incorrect or missing information.

What Is Enterprise Networking?

Enterprise networking is a system of organization-wide data communications that supports you and your computers wherever you may be in a company. This definition implies some important capabilities: that an enterprise network must serve all levels in an organization and that it must not be sensitive to location.

For example, if only senior management is on a network, then it's a senior management network. If only the headquarters building is on a network, then it's a headquarters network. Enterprise networking covers the entire organization or enterprise.

Because of their scope, enterprise networks have a few common characteristics, most notably their tendency to concentrate data in some form of central storage. They hold important corporate data in a common location where the data can be protected and archived so that the loss of the central storage facility will not destroy that data.

In addition, enterprise networks are ordinarily made up of groups of LANs linked by a communications medium. The exact nature of this medium depends on the company.

Note that enterprise networks need not have large numbers of users, nor do they have to span continents or oceans. Although many such networks do support a large user base and cover great distances, there are also enterprise networks that connect relatively few users over short distances. The determining factor is the size and breadth of the company.

I know of a Mexican restaurant chain in the Washington, D.C., area that has a small LAN at each of its two locations. The LANs are linked by a dial-up line so that they appear to be a single LAN. This allows accounting reports, requests, and even recipes to be moved between the two locations. Small? Sure, but it is as much an enterprise network as Microsoft's 18,000-node network, which almost stretches around the world.

In fact, there's nothing in its definition that requires an enterprise network to exist in more than one building. It's even true that you can reach a point at which an enterprise network becomes so small that it is indistinguishable from any other small LAN. It all depends on the size of your business.

Centralized archival storage is a top selling point for an enterprise network. Nothing strikes fear into the heart of a manager more than the thought of losing valuable information or having that information compromised by allowing an unauthorized person to gain access to it. Because enterprise networks allow you to manage corporate data centrally, they enable you to deal with these problems.

The most important element of centralized data management for most businesses is the ability to back up all the company's file servers from a central point. With individual hard disk drives, you can't ensure that backups are run, and for the most part, they aren't. It's not unusual in a decentralized scheme to find the financial information necessary to run a $100,000,000 business scattered across a variety of $300 hard disk drives.

Most companies use the same techniques to back up their enterprise networks that they once used to back up their mainframes: They take incremental daily and full weekly backups of all information on the network. The process can be labor-intensive, but it ensures that the data is protected. Likewise, corporate networks let you implement some means of access security, lessening the chance that the competition will walk off with company secrets.

An enterprise network, then, is a network that provides connectivity to an entire organization. It is available at all levels within the company, and in most cases, it provides connectivity beyond the limits of individual LANs. The enterprise network provides connectivity between departmental LANs as well as a means to access corporate resources, such as a centralized archival facility. It can also provide access to other central resources, such as a mainframe.

What's the Big Deal?

Why is enterprise networking suddenly so important? Because, although most of the technology necessary for enterprise networking has been around for a while, it's only in the last two or three years that the technology has become an important factor in the growth of the use of personal computers and LANs in corporations. As you might expect, there is no single reason for this sudden emergence.

The most pervasive factors in the sudden growth of enterprise networking are the current economic environment and the related rising cost of labor. These elements have changed the way many businesses are organized, which affects the way you work.

In addition, after over a decade of growth, personal computers and LANs have reached a critical mass sufficient to support enterprise networking using existing equipment. And although the technological support for enterprise networking has been available for a while, some critical technologies have matured only just recently to the point where enterprise networking is now practical rather than simply possible.

The combination of economic factors and changes in the way you work is the key element in the acceptance of enterprise networking as an attractive answer to the challenges faced by corporations. The rising cost of labor affects this acceptance in two ways: First, companies must pay more for the time you work; and second, most companies are under increasing pressure to accomplish more work with fewer people. You must be both productive and flexible.

As a result, companies are finding that they must adopt something resembling a matrix organization, where you may find your work assignments changed often. One way to make these changes easily is to form workgroups that don't require the participants to be in each other's physical presence. The corporate network's ability to support such workgroups makes matrix management possible in many companies.

In addition to changing how you work, economic forces have also drastically lowered the price of computer hardware. Today, you can have a computer that is more powerful than the mainframes of yesteryear for less money than an electric typewriter used to cost. This revolution in inexpensive computing power is the second key factor in the sudden growth of enterprise computing.

The growth of personal computing began a decade ago, and although it started slowly, many of the microcomputers purchased by companies in the early 1980s are still in use. Coupled with the more powerful machines of today, these older microcomputers create an enormous base of individual-computing resources.

While this personal computer base improves individual productivity a lot, it is also a key factor in the network: It is the foundation for a corporate enterprise network. Personal computers, whether they are early models or more recent ones, all have a role to play in the enterprise network. The fact that they are already in place means that you can build the network around them.

The last key factor is a series of critical enabling technologies that make enterprise networking economically justifiable. These technologies include new bridge technologies, reasonably priced high-speed data networks, high-speed modems for dial-up use, new transmission media, practical methods of centralized data storage, and new ways to manage huge networks. Although enterprise networking is possible without some of these technologies, the arrival of each of them has made the concept more attractive.

Taken together, these factors transformed enterprise networking from an interesting theory to a process that could be achieved within a few months. Suddenly, companies had to have networks, and they had to network their entire business. Nothing less than an enterprise network would do.

New Technologies

Ultimately, the success of a technology in the computer business is dependent on two elements: First, the technology must fill an identifiable need; and second, it must be practical. The technology must be easy for companies obtain; in other words, it must be both possible and affordable. A company also needs some assurance that it will be successful in implementing the new technology. This is why companies don't usually purchase their own communications satellites—it is certainly a possibility, but it's expensive, and success isn't assured.

The need for enterprise networking is based on the belief that it will fulfill a requirement that exists in a particular corporation. A series of technologies became available within a few months' time and in such a way that they helped to create a ground swell of acceptance of enterprise networking. Interestingly, most of the technologies involved were versions of earlier technologies that became easier to use and/or much less expensive to buy. In some cases, they expanded the use of current technology into new areas. For example, network-bridge technology changed so that the products became both less expensive and better suited to corporate use.

A bridge is a device that lets you connect two networks and operate them as if they were one. One common type of bridge lets you connect two Ethernet LANs via a Tl line so that the LANs seem to be directly connected. Other bridges let you link LANs through microwave links, lasers, or dial-up telephone lines.

Originally. bridges were expensive and/or highly proprietary. If you bought an Ethernet-to-Tl bridge, you'd pay tens of thousands of dollars and might find that it passed Ethernet packets from only one manufacturer's equipment.

Now, with bridges like the MLB/1600 series from Microcom, the bridge is an expansion card that mounts on any IBM AT compatible. This bridge passes anything that comes across the Ethernet or the Token Ring and does so over a variety of transmission media, including Tl, fractional Tl, or dial-up lines. You can even create a temporary bridge when you need one.

Previously, gateway technology that let your LAN access an external computer was little different from bridge technology. The gateways that were available to businesses were very limited. Cost was not so much of a factor—when you're paying $500,000 for a mainframe, a couple thousand dollars one way or the other for a gateway isn't much of an issue.

The problem was that mainframe manufacturers had little incentive to produce LAN gateways, so they left it to the LAN companies to develop them. Important items, such as Systems Network Architecture gateways, have been around for a while, but less popular items, such as gateways to the Honeywell DPS-6, were a lot longer in coming. Because companies wanted to make the most of their investment, they wanted the company network to provide access to their mainframes and to the personal computers. Where such connectivity was difficult, corporate acceptance of LANs slowed.

In addition, the leading LAN developers were sometimes slow in providing a wide range of connectivity. Novell, for example, is only now providing full TCP/IP connectivity. Connectivity to VAX computers through LANs is fairly recent for everyone except DEC. IBM, meanwhile, was so busy with new connectivity solutions that everyone else was caught off balance for a while.

The gateway technology has improved as well. Current technology now provides gateways that are software running on a computer on the LAN. Others are limited to one expansion card for a PC. Gateways remain expensive, but cost is less of an issue because most companies only need a few of them.

New Ways to Move Data

Bridges and gateways benefit from access to high-speed data communications. This is an area that has recently exploded. Once, high-speed remote data communications usually meant a Hayes 2400-bps modem; now you can achieve effective throughput at speeds of up to 38.4 kilobits per second with V.42bis modems.

Communications throughput on dial-up lines now rivals that of leased data lines. Branch offices can now establish communications with each other at costs much lower than previously. The speed is slower than a 56-Kbps data line, but in many cases, it's plenty fast enough.

Packet-switched public data networks also play a significant role in making enterprise networking function over

Enterprising Storage

Ten years ago, the ISO seven-layer Open Systems Interconnection model helped standardize the logic of communications by providing a common groundwork for communications protocols. Currently, the IEEE is working to define a comparable logic for network-storage interchange. The two-year project is supposed to generate a five-layer model that will provide a structure for solving the data management problems introduced by network computing and rapid growth of on-line information.

Who Needs It?
The shift from centralized time-sharing computing to network computing is in full force. Known by various terms (e.g., client/server, workgroup, downsizing, or rightsizing), network computing provides quick response times and easy-to-use visual interfaces. That's the good news. The bad news is that the transition has introduced greater complexity into the data management process.

In many companies, individual sites now include computers from any different vendors. These sites used a centralized, integrated service for managing distributed data that permits the same degree of control that centralized computing offers.

Two economic factors are having a great impact on the management of distributed data. First, the cost of small hard disks has fallen so dramatically that providing all the local storage you need is not a very expensive proposition. In the past, desktops, as much as workstations, were diskless for economic reasons—hard disk drives cost a lot of money. Today, diskless workstations are still around, due primarily to the lack of acceptable storage management standards.

Second, mass storage in the form of optical or tape robotic devices is most economical when you buy it with large capacities. When you replace a mainframe with a mass-storage device on a network, you need a common network service to distribute and share the low-cost storage throughout the network.

Defining the Model
The IEEE Mass Storage System Reference Model, which is currently being developed by the IEEE Technical Committee on Mass Storage Systems and Technology, provides a framework for describing the functionality required by mass-storage systems. By providing a consistent set of concepts and terminology, the reference model lays the foundation for the development of standard mass-storage architectures and interfaces. It does not dictate a specific implementation architecture.

The reference model divides a mass-storage environment into the following logical entities.
• The *Bitfile Client* presents an application-oriented storage abstraction, defining such concepts as files, images, directories, and file attributes, as well as access control.
• The *Bitfile Server* provides the storage needed to implement the Bitfile Client's abstract view. The Bitfile Server manages objects called *bitfiles*, which hold uninterpreted data and attributes and are identified by globally unique identifiers called *bitfile IDs*. The bitfile is the core component of the reference model.
• The *Name (Attribute) Server* provides a mapping between application-oriented names and the IDs of the bitfiles used to hold the storage service's data.
• The *Storage Server* provides a set of perfect (defect-free) logical volumes that the Bitfile Server uses to hold bitfiles. These logical volumes may have associated properties, such as size and location.
• The *Bitfile Mover* is a data-movement service that the Bitfile Client and Bitfile Server use to transfer large quantities of data between logical volumes and applications. It's currently the most disputed component of the model.
•The *Physical Volume Respository* manages the physical media used to implement logical volumes. Its tasks include physical volume indentification, access control, jukebox control, and physical device access.
• The *Site Manager* provides tools for monitoring and controlling the actions of the other services.

Epoch Systems (Westborough, MA) has developed the Infinite Storage Architecture based on the emerging IEEE mass storage model. ISA was developed to provide reliable, transparent access to on-line storage. It was designed to provide automatic data management that requires little human intervention to permit all data to be on-line at all times and to let distributed systems enjoy the same reliability and ease of use that centralized data management systems do.

Bitfile Basics
The bitfile is the lowest common denominator in the ISA. Each bitfile is named and accessed with unique 128-bit network-wide identifiers. A name-generation service lets different storage servers create bitfiles, and it guarantees their uniqueness. Bitfiles are managed by the Renaissance InfiniteStorage protocol,

---

long distances. Although it has always been possible to connect two LANs over a service like Tymnet, the results have not been satisfactory in most cases. Now PDNs support a variety of services (e.g., 56-Kbps, frame-relay, and fractional-Tl lines). So you can establish a link between two offices at a speed that provides useful throughput.

Other types of data communications are growing as well, and the growth of enterprise networking is creating a niche for them. Several companies are selling infrared lasers designed to send LAN transmissions between buildings in the same metropolitan area.

Likewise, there are several sources for microwave transmission equipment designed to be used with networks. Like lasers, a microwave connection is only useful for line-of-sight operations.

Other solutions include central-office LANs, offered by Bell Atlantic, among others. This technology lets a Centrex switch handle LAN data as well as voice telephone traffic. If that's not what you want, the telephone company will install fiber-optic cable in its phone conduits—as long as you're willing to pay the price.

Centralized Data

Changes in how data is handled have paralleled the changes in communications technology. The most significant is the growth in LAN-based database servers. Coupled with the quick acceptance of Structured Query Language (SQL) as a universal database language, these servers give companies a valid reason to use LANs for something besides E-mail and backup. In fact, database which is implemented with remote procedure calls.

Higher-level network-storage services (e.g., hierarchical storage and backup) can also be implemented using the bitfile service. You can think of the bitfile service as a network-wide, logical-storage name space. And the machines that support the bitfile service are called storage servers, as opposed to file servers.

Most important, ISA provides bitfile service to all file system and database servers and services on the network, thereby creating a common storage service. One mass-storage device can provide storage for a database and a file server.

Epoch Systems has already applied this technique to very large UNIX networks. Each workstation and file server runs a copy of a migration service and a bitfile client. Together, these services seamlessly integrate with existing storage services (Network File system and UNIX File Systems) to provide networking hierarchical storage.

Mass Storage and Archiving
Volume Management and Automated Library Management services are Epoch Systems' implementation of the IEEE Physical Volume Repository. Although mass-storage robotics are very similar in most implementations, the lack of robotics standards makes integration time-consuming as well as proprietary.

A standard abstract interface for robotics integration can lower costs, reduce engineering burdens, and free you from proprietary solutions. The effects of such standards have been shown in Epoch's labs, where SCSI-2-based robotics for disk and tape products have been integrated in two weeks.

When data is archived, it is removed from your immediate reach. With the inexpensive storage that's available, it's now possible to keep all your information on-line. Keeping information accessible at all time requires new solutions to old problems—most notably, how to perform a backup and how to locate the data.

In traditional systems, the time required to complete a backup is a function of how much data is involved. Because old data is usually removed or deleted from the system, the accumulation of new data dominates backup.

ISA-based systems are just the opposite. Because information is never logically removed from the system, less-used but nonetheless valuable data soon dominates the bulk of on-line storage. The time needed to complete a consistent backup needs to be a function of the data that has changed, not of the data that must be kept on-line.

Epoch Systems developed patented baseline-backup technology to meet the objectives of a system where all information is on-line all the time. For example, baseline backup will perform consistently whether the 30 gigabytes of changed information resides on a 100-gigabyte or 2-terabyte storage server.

As the moment of on-line information reaches immense capacities, it becomes more effective to constantly keep track of what information has changed rather than to periodically search the entire storage space.

The Office Is the Enterprise
The office of the future will feature human-oriented, data-intensive multi-media applications. Information created at the desktop will flow throughout the networked office. Even novice users will be creating and using computer-based information.

In such a system, you can't expect individuals to be responsible for the preservation and maintenance of office information. Offices of the future will plug into an integrated network that provides transparent access to storage services. Individual desktops will access this storage on demand, and backup and archive tasks will be automated. You will access information through your applications. It will be up to the storage services to ensure that information is reliably managed and moved on demand to the appropriate storage level.

As the workgroup concept spreads and more immediate problems (e.g., network management) recede, you will come face-to-face with the problems of storage management over a network. The IEEE model—and architectures based on it—promises to do for network storage what the ISO reference model has done for network communications.

—By Gregory G. Kenley
Vice President of Strategic Planning Epoch Systems
Westborough, MA

---

servers give a corporation much of what it had in mainframe days—control over its data—while giving you the freedom you have come to expect with personal computers.

Database servers are moving beyond simply handling databases; they are becoming applications servers. Although they still store data, the queries and commands they satisfy are coming from non-database programs. These programs can be anything from image processors to spreadsheets.

Information is easier to track when it's on one database server than when it's spread over many hard disk drives on individual computers. So the company is more likely to pool information in a place where it can be accessed over a LAN, making the same information available to everyone.

Another factor in the growing popularity of the centralized data server is that it gives continued life to mainframe computers. With an enterprise network, you can access data from a DB2 database running on a mainframe as easily as you can access data from a SQL Server database running on a Compaq Systempro. The mainframe can continue to do the things it does best while the data it stores becomes available over the network.

Managing the Network

Once the network is functioning, it must be kept running. The problems encountered in operating a network are legion. They include cable faults, traffic congestion, software version control, and component failures. Until recently, managing a network meant hiring many people to baby-sit the network, offsetting much of the cost advantage of enterprise networking. Now technology can overcome this problem.

Several companies, including Cabletron and Synoptics, that make hardware have developed comprehensive network management software that works with their own hardware. A single workstation can manage Cabletron's or Synoptics' hubs, bridges, routers, or network interface cards. In most cases, these packages run under Windows or OS/2 and take full advantage of the graphical environment to make the software easy to use and the network easy to control.

In addition to supporting proprietary systems from adapter and hub makers, many manufacturers also support management systems like IBM's NetView (a mainframe-oriented network management system) and Simple Network Management Protocol. SNMP started out in UNIX networking and is becoming widely accepted in the networking world. In fact, as more networking devices add support for SNMP and as companies that used to rely on proprietary network management add SNMP support, the industry is getting closer to supporting SNMP as the single standard for enterprise network management support.

A Bright Future

There is no question that corporations will embrace enterprise networking. As the millennium approaches, the trend toward enterprise networking will become a rush, and the rush a flood. The driving force will be the desire of companies to control their information while promoting communication among their people and making them more productive. The pace will quicken as companies discover that the return on their investment is better when they add computer capacity through networking instead of by buying more mainframe computers.

The road to enterprise networking is not open and clear, however. There is a sad lack of standards in some critical areas. (For a look at an evolving standard for network storage, see the sidebar"Enterprising Storage.")

Where standards exist, many vendors insist on providing proprietary products. For example, the bridges at both ends of a communications line must be from the same vendor. You must pick one vendor for each line you use. Fortunately, the bridges that connect LANs are moving away from these requirements.

Network management is closer to standardization. As enterprise networks expand, management becomes critical, and companies are demanding standards. By the end of the decade, all enterprise-network management solutions will probably support SNMP, perhaps side by side with a vendor's proprietary solution.

Overall, the future is bright for enterprise networking and the industries related to it. Businesses need networks that connect all their operations. These networks provide a means for using database servers, E-mail, group productivity software, bridges to other systems, and gateways to the company mainframes.

The industry has reached critical mass in enterprise networking. The concept will spread and drive growth throughout the small computer industry. All that remains is to see how fast the growth will be and where it will take us.■

Using SNA to Link LAN to Host

In this report:

Datapro Summary

SNA gateways have been useful in providing quick and easy fixes for LAN-to-mainframe links; however, they have their share of network management and throughput limitations. For high-volume environments, directly running SNA might be a more effective, albeit more complicated, solution. This report presents a series of connectivity scenarios that illustrate proven, effective ways to connect LANs and mainframes. An implementation checklist for embracing an SNA integration project is also included.

It used to be so simple: If you wanted to connect a personal computer to a mainframe, you installed a 3270 emulation card and made your desktop system act like a dumb terminal. Until recently, IBM's Systems Network Architecture did not allow for deviation from its hierarchical model of computing and communication, and the workstation was strictly a data entry device as far as the mainframe was concerned. Then came LANs.

IBM was slow to integrate LANs into its data communications strategy, let alone to provide LAN-to-mainframe connectivity. As a result, third-party vendors such as Digital Communications Associates Inc., 3Com Corp., and Novell Inc. brought out SNA LAN gateways. These products, which are either black boxes or workstations dedicated to executing gateway software, allow LAN workstations to share 3270 emulation capabilities and eliminate the need for each computer to have an emulation card.

At their core, however, 3270 gateways are kludges in that they fool the mainframe into thinking it is talking to a 3X74 controller with terminals hanging off it. If it's not nice to fool Mother Nature, it's even worse to fool IBM's mainframe communication software, Virtual Telecommunications Access Method (VTAM). In general, gateways tend to be unreliable and suffer from poor network management since manufacturers do not always implement management capabilities that can interoperate with SNA network management tools. In addition, gateways often become performance bottlenecks because low-powered personal computers are used to run the gateway software. The high volume of I/O a gateway must handle puts considerable strain on a personal computer.

As a result, many organizations are going native and throwing out SNA gateways in favor of using SNA protocols to connect personal computers on a LAN directly to their mainframes. They are accomplishing this using either a local or remote communications controller. A local controller attaches directly to the mainframe while a remote controller attaches via a synchronous data link control (SDLC), token ring, or T1 connection. Although running SNA over LANs wasn't an option five years ago, IBM has finally brought together the hardware and software to allow LAN-to-WAN integration under SNA.

This Datapro report is a reprint of "Using SNA to Link LAN to Host" by Philippe Byrnes, pp. 35-43, from *LAN Technology*, Volume 6, Number 12, December 1990. Copyright © 1990 by M&T Publishing, Inc. Reprinted with permission.

IBM's Communications Strategy

As many LAN managers have discovered, mainframe data communications is a very different ball game, both in terms of the options you have in running SNA to a LAN and in the complexities encountered along the way. This report outlines the various ways to interface a LAN with a mainframe computer installation that is running SNA communication software. After an overview of essential features of IBM's data communications products and protocols, I discuss criteria that can help you determine the best way to connect your LAN to an IBM mainframe, including a number of scenarios that cover most contingencies. I've also included a checklist that can help minimize problems along the way and make connecting your LAN to an SNA mainframe a relatively painless operation. If you have a high volume of traffic and require high reliability in your LAN-to-mainframe connection, you should consider running SNA natively.

Many a data communications veteran has said that half the battle in understanding IBM communications (and SNA in particular) is mastering the terminology. IBM has a tendency to use its own lingo rather than the terms the rest of the industry uses. For example, where other vendors speak of wide and local area networks IBM speaks of enterprise and establishment networks. Unfortunately, the two terminologies do not map one-to-one. Enterprise should not be taken to mean remotely attached resources nor should establishment be translated to locally attached resources.

In addition to having its own language, the IBM communications world covers a lot of territory. Since its introduction in 1974, SNA has been the protocol of choice for constructing large wide area networks. Notwithstanding the popularity of TCP/IP in the academic and engineering/scientific communities, SNA overwhelmingly dominates commercial environments because its protocols allow for better routing, fault tolerance, and network management than any alternative protocols. Large SNA networks can easily have more than 100,000 devices (computers, terminals, and printers) attached to them and require a support staff of hundreds of systems programmers and network operators. On the other hand, it is possible to have small, relatively simple SNA networks. For example, Advanced Peer-to-Peer Networking, a variant of SNA designed for networking low- and midrange systems such as personal computers and minicomputers, requires minimal support and user sophistication.

In many companies, because LANs were outside the scope of the main SNA network and outside the control of the Information Systems staff, the mainframe and LAN worlds have largely been isolated from each other until recently. Network managers and operators on both sides lack an understanding of each other's tasks and challenges. LAN administrators who think NetWare is complicated have yet to discover the joys of VTAM and Network Control Program, the software packages that are IBM's networking workhorses. SNA mavens, however, shouldn't be too cocky, for much of SNA's recent evolution has been in direct response to innovations pioneered in the LAN world. LANs brought us such non-SNA concepts as peer communications and any-to-any connectivity, in which computers of any size that support SNA can be linked. LANs also brought us the concept of dynamic networking, in which users can be quickly added and deleted from the network and a link can be added without requiring the whole network be brought down and back up again. IBM has been forced to add these and other LAN features to SNA and, in so doing, change it drastically from what it was 10 years ago.

Controllers: The NIC Equivalent

LAN administrators used to popping a network interface card into a personal computer and attaching it to a LAN often don't realize that it's not possible to drop an adapter card into a $5 million mainframe and start up your network. IBM's System 370 line, including the 4381, 3080, and 3090 families, and the recently announced System 390 machines, have communications channels, but these are used only to connect specialized I/O controllers such as data communications processors and devices such as tape and storage units. IBM's mainframe architecture employs these channels for two reasons: Specialized computer architectures are more efficient and you want to offload the mainframe processors because mainframe MIPs are the most expensive around.

Only a few IBM hosts, including the AS/400 and select IBM System 3X0 machines (the 4361 and some ES/9370 and ES/9000 models) do not use I/O controllers and do not have channels. Instead, these host models come with integrated communications adapters that allow the direct attachment of SDLC and token-ring interface cards.

This fall IBM announced a 10-Mbyte/sec fiber-optic Enterprise System Connectivity (ESCON) serial channel. The fiber-optic technology provides several new capabilities, including higher bandwidth (roughly double that of existing copper-based parallel channels) and the ability to run the channel much greater distances. ESCON allows channel-attached controllers to be up to 5.6 miles from the mainframe. As a consequence, many LANs that otherwise might have required a relatively low-speed remote connection to the mainframe due to the distances involved can now be attached directly to the host at channel transmission speeds.

By contrast, the older types of channels can only be run several miles by using the 3088 channel extender. In addition to the longer distances that fiber cable can be run, another plus for the fiber connection is that fiber cable is more lightweight and compact than the thick, parallel copper cables used to date.

An IBM mainframe is attached to a communication link or a LAN via one or more communications controllers. Controllers are machines dedicated to managing the inbound and outbound data communications, so the mainframe does not have to be interrupted constantly to execute I/O tasks. In this sense, a controller acts as a front-end processor. These I/O controllers take the interface cards, such as the token-ring interface coupler, SDLC, or T1, used to link devices on an SNA network.

IBM has three families of communications controllers which it calls Enterprise, Establishment, and Interconnect controllers. Each off-loads communications tasks from the host in different ways (see Figure 1).

- **37X5/3720 Enterprise Controllers:** These are also called front-end processors or simply communications controllers. Large, expensive ($500,000+) machines, these controllers run the Advanced Communications Function/Network Control Program responsible for executing most SNA network operations. Products include the 3745, 3725, and 3720. Local versions of enterprise controllers connect to the host via a channel and support SDLC, token-ring (with an interface coupler), or T1 connections out to the network. The remote versions can support SDLC, token-ring, and T1 connections on to both the network and the mainframe.

- **3X74 Establishment Controllers:** Formerly called cluster or 3270 controllers, these are small- to medium-size machines that attach 3270 terminals and printers to a host. Products include the 3174 and 3274. A local establishment controller connects to the host via a channel and to terminals and printers via coaxial or twisted-pair cable. Remote versions of 3X74s support coax connections to terminals and printers and token-ring (using a network token-ring interface), SDLC, or X.21 links to the host.
- **3172 Interconnect Controllers:** The newest class of IBM controller, these are dedicated 80386- or 80486-based Micro Channel Architecture machines that act as high-speed bridges that can be used to connect a mainframe to a LAN or one LAN to another. The 3172 currently supports token-ring, token-bus, Ethernet, IBM PC Network, T1, and FDDI networks. 3Com recently announced it will be supplying Ethernet adapter boards for the 3172.

All of these controllers have remote versions that are not channel attached. Rather, most support an SDLC, token-ring, or T1 connection on the network side and attach to an enterprise controller on the host side. Channel-attached controllers have the advantage of higher speed over the remote versions. The channel connections operate at speeds in the range of 8 to 32 Mbits/sec (up to 80 Mbits/sec with ESCON) while the remote controllers are usually linked with SDLC connections that range in speed from 9.6 to 256 Kbits/sec.

Independent vs. Dependent LUs

The procedural interface used to access an SNA network's services is much more varied and complicated than that presented by TCP/IP or Novell Inc.'s SPX/IPX. SNA's service access points are called logical units or LUs. Basically, LUs are the software implementations of the upper layers of the SNA protocol stack, which is found in all programmable IBM computers. In the mainframes, the LUs are either in application subsystems, such as CICS (Customer Information Control System), IMS (Information Management System), or TSO (Time Sharing Option), or in the underlying operating systems such as MVS or VM. The Novell analogue to LUs is the SPX portion of NetWare in the shell and server software.

Although IBM has defined half a dozen types of LUs, for all intents and purposes only LU 2 and LU 6.2 are important today. LU 2 supports 3270 terminals and the mainframe software that communicates with these terminals. LU 2 was designed with the assumption that all the processing power was in the mainframe, so the mainframe and outboard logical units function in a hierarchical, master/slave fashion. The LU in the mainframe contains almost all the intelligence while the LU in the terminal (actually the 3X74 controller) performs very limited tasks. Until the mid-1980s, this was a valid approach since most of the devices attached to the mainframe were terminals and printers with very limited amounts of memory. Because most mainframe software in use today was written using the LU 2, most users still access the mainframe using LU 2 protocols.

LU 6.2, on the other hand, is the long heralded successor to LU 2 and the other LUs. It allows peer communications rather than the master/slave flow control of LU 2. Although it has been around for more than six years, LU 6.2 is only now being fully implemented throughout the IBM product line as part of Systems Application Architecture.

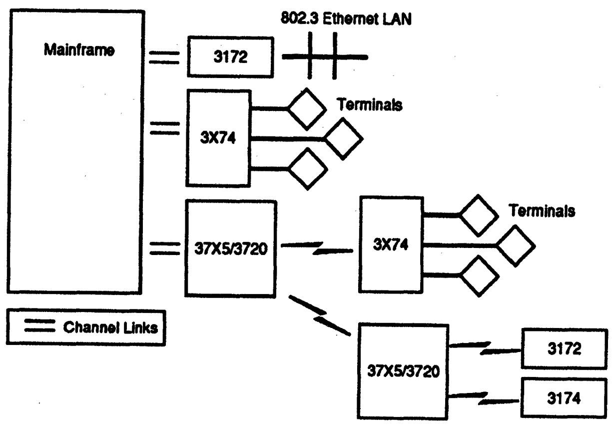
LU 6.2 is called the independent LU while LUs defined earlier are referred to as dependent LUs because of the severe restrictions they impose on one or both partners in a session (see Figure 2). Not all IBM products have implemented the features necessary to support sessions between independent LUs—an important fact to keep in mind when you are selecting a LAN-to-mainframe connection option.

An Implementation Checklist

If you follow the checklist below, you may avoid some common mistakes made by network managers when tackling an SNA integration project for the first time. The best thing is to get all the information you need upfront and secure the cooperation of everyone whose help you will need.

- Identify user requirements. Are your workstations going to access host software primarily as 3270 terminals or are more sophisticated, client/server applications to be used or developed? What are the response times and the percent of uptime needed?
- How far apart are your LAN and mainframe? If they are within local attachment distance, can cable be run without problems? What about rights of way?
- Does your mainframe have any unused channels? If not, then local attachment may not be an option even if the distances involved would allow it.
- Whether you plan to use remote connections or local ones, if you are using existing front-end processor(s) check them to see if they have room in the backplane to attach your line interface or token-ring interface cards. Are the scanners on the host fully saturated or is there unused capacity? If they are saturated, you must either upgrade the scanner(s) or buy another front-end processor. (A scanner is a processor that reads all the interfaces to a 3745 to see if they have data coming in. Different types of lines use different amounts of scanner processing power, so you need to know if the scanner on the host you wish to attach to can accommodate your connection.)
- Make sure you have a release of the communications software that has the features you need, e.g., token-ring support, PU 2.1, etc. Do not assume your mainframe and front-end processors are up to date. Many organizations have very old copies of VTAM and/or NCP still in use.
- Secure the cooperation of the system programmers responsible for generating new VTAM and NCP tables. These people can be your best friends or your biggest impediment. Coordinate cutover activities early to give the host programmers plenty of time to make the changes you need.
- And remember which is the cart and which is the horse. The user requirements should drive the process, not the other way around.

Figure 1.
IBM's Three Families of Communications Controllers



IBM's enterprise, establishment, and interconnect controllers allow different types of equipment to be connected to host computers.

Mainframe Connectivity Issues

It is an axiom of good system design that requirements should drive capabilities. A network's design should be driven by the traffic the network is to carry, not the technology available. The requirements for the LAN-to-mainframe connection in your installation may vary considerably from those in another company, so it is impossible to give a generic paradigm of putting things together.

Broadly speaking, there are four requirements that go a long way toward determining the optimal connection between your LAN(s) and your mainframe(s). These are the relative locations of the LAN(s) and the mainframe(s), the protocols used, the price/performance desired, and whether the applications that are to be accessed in the mainframe and/or on the LAN will work as peers or in a master/slave fashion. The last may be the most difficult requirement to estimate, for the applications used today may not be those used in two years.

The old saying about real estate—that the three most important things are location, location, and location—is equally true when determining the best way to connect an SNA mainframe to a LAN. If the two are in the same building, the best thing is to locally connect the mainframe to the LAN using a channel-attached controller.

Even in a campus, where the LAN and the mainframe may be several miles apart, channel-attached controllers can be used either with the new ESCON channels, or the older channels can be used with channel extenders to go up to five miles. If you need to connect a mainframe and LAN over longer distances, you can use a pair of enterprise or interconnect controllers with T1 adapters and extend the connection as far as your common carrier will let you run a T1 line. Alternatively, remote controllers can be used to connect the mainframe and LAN. In this configuration, the controller(s) would have LAN and SDLC interfaces, with the downstream link going into a wide area SNA network and then back to the mainframe.

Another thing about mainframe connectivity products: they aren't cheap. Your mainframe-to-LAN connection might require you to purchase a front-end processor. A small 3745 costs approximately $100,000, while a large one can tip the scales at over $1 million. NCR Corp.'s Comten division, Fujitsu America Inc., and Amdahl Corp. are among the third parties that offer IBM-compatible communications controllers, but there is the danger that IBM will make its competitors' hardware obsolete by changing interfaces.

Each connection alternative—local or remote, enterprise or establishment controller—has pros and cons in terms of cost, performance, and network management capabilities. Although an enterprise controller is expensive, it handles many more SNA functions and has more management capabilities than an establishment controller. For example, an enterprise controller handles the translation of the 6-byte address used by hosts into the 1-byte address used by terminals and other devices. This translation, called the boundary function, is not handled by an establishment controller, so it must be done by the host, which puts an additional burden on host processors. You must decide whether the benefits of an enterprise controller, an ESCON, or a channel extender, for example, are worth the price.

The last issue to consider is the host and/or workstation applications you are connecting. As I noted, SNA networks support two types of applications, peer and hierarchical, connected by peer LU 6.2 or hierarchical LU 2 sessions, respectively. Most organizations do not yet have a large demand for LU 6.2 connectivity because most mainframe applications still use LU 2.

The temptation is to plan for an LU 2-based mainframe-to-LAN connection. It's less costly and is based on stable, existing technology. The danger is that you will outgrow it sooner or later, and it may be sooner. LU 6.2 is IBM's strategic communications protocol, allowing a session between any two computers. With LU 2 you are restricted to dependent LUs, and your workstations can use LU 2 sessions only to communicate with the mainframe(s), not with each other. (When operating hierarchically, SNA requires that a primary LU initiate a session with a secondary LU—the reverse is not possible—and secondary LUs cannot communicate with each other. LU 2 does not support a primary logical unit outside the mainframe; only LU 6.2 does.) What all this means for mainframe-to-LAN connectivity depends on whether you want to access the mainframe as a 3270 device or as an intelligent computer. If you want the latter, you are limited to using an enterprise communications controller running Network Control Program Version 5.2 or later because earlier versions of NCP do not support the physical unit 2.1 components. A physical unit is the agent for managing SNA-defined resources in a node. PU 2.1 is necessary for processing LU 6.2 session bind requests received from independent LUs in computers on the LAN.

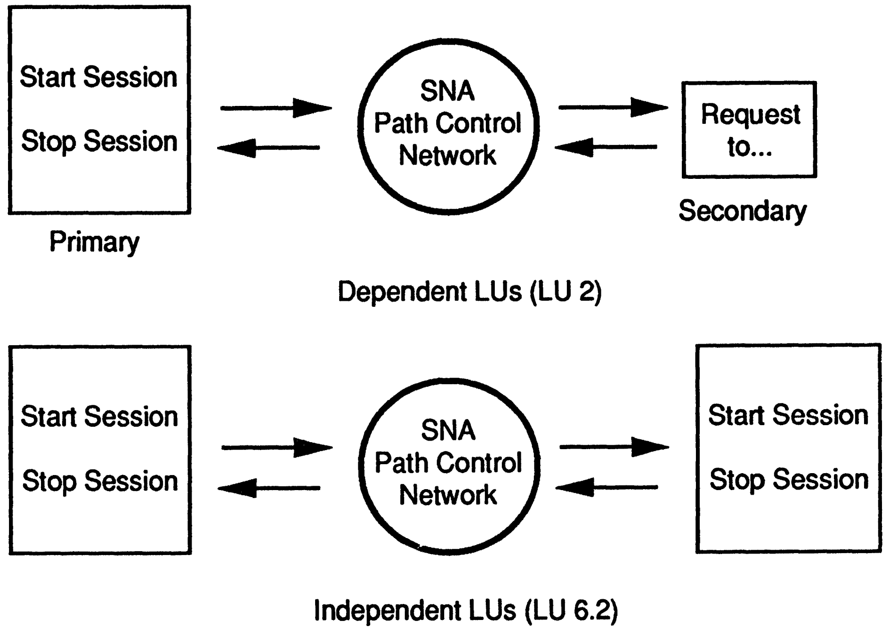
Choosing a Connection

Connecting a LAN and a mainframe is a lot easier if the LAN is a token ring because IBM has spent the past five years working to integrate token-ring networks into SNA. In the mid-1980s, IBM selected token ring (IEEE 802.5) as the local area network protocol it would support across its product line and include as an alternative to SDLC as an SNA data link protocol. Controllers such as the 3745 and 3174 can be connected directly to a token-ring LAN, for example. However, you can only use a 3172 to link an Ethernet (802.3), token bus (802.4), or IBM PC Network LAN to a mainframe.

Although there are several requirements to consider, you basically have two choices to make when connecting

Figure 2.
Dependent and Independent LUs

Sessions between dependent LUs are hierarchical in nature; only a primary LU such as a host can initiate a session whereas a secondary LU such as a terminal can respond only when polled. By contrast, sessions between independent LUs are peer-to-peer, where either mode can start or stop a session.



your LAN to your mainframe: Will you attach the mainframe locally or remotely, and will you support independent and/or dependent LUs? If you opt to support your networked personal computers as independent LUs, you will need to run IBM's APPC/PC on MS-DOS machines or OS/2 Extended Edition (or equivalent products) to get LU 6.2 support. To support dependent LUs, you need to run IBM's Personal Communication 3270 program on DOS machines or OS/2 Extended Edition (or equivalent products). Depending on which options you choose, you will end up with some variation of these four basic scenarios (see Figure 3).

Scenario 1: Mainframe on WAN, Hierarchical Connectivity Using Third-Party 3270 Gateways
The status quo for connecting a mainframe to a LAN is to use a gateway that emulates a 3X74 cluster controller. As I noted earlier, the gateway can be a single point of failure for all the stations on the LAN that need to access the mainframe. Another disadvantage is that NetView's network management capabilities stop at the gateway and cannot see the nodes that lie beyond on the LAN.

Scenario 1a: Mainframe on WAN, Hierarchical Connectivity Using 37X5/3720
The advantage of this configuration over an SNA gateway is that the throughput between the mainframe and the LAN is enormously greater. So is the cost.

Scenario 2: Mainframe on WAN, Peer Connectivity Using 37X5/3720
The driving requirement in this configuration is peer connectivity. The 3X74 controllers do not support physical unit 2.1, so if you want peer (LU 6.2) sessions you must use 37X5/3720 front-end processors. The solution is to use a remote front-end processor with a token-ring adapter and connect it over long haul SDLC or T1 links to a locally attached front-end processor.

Scenario 3: Mainframe on LAN, Hierarchical Connectivity Using 3174
In terms of quick, dirty, and cheap, this configuration is probably the winner. By using a channel-attached 3174 with a token-ring adapter instead of a 37XX controller, the cost is greatly reduced compared to the following scenario. On the downside, the mainframe must execute the boundary function tasks, such as converting packet identifier formats, that NCP would otherwise perform.

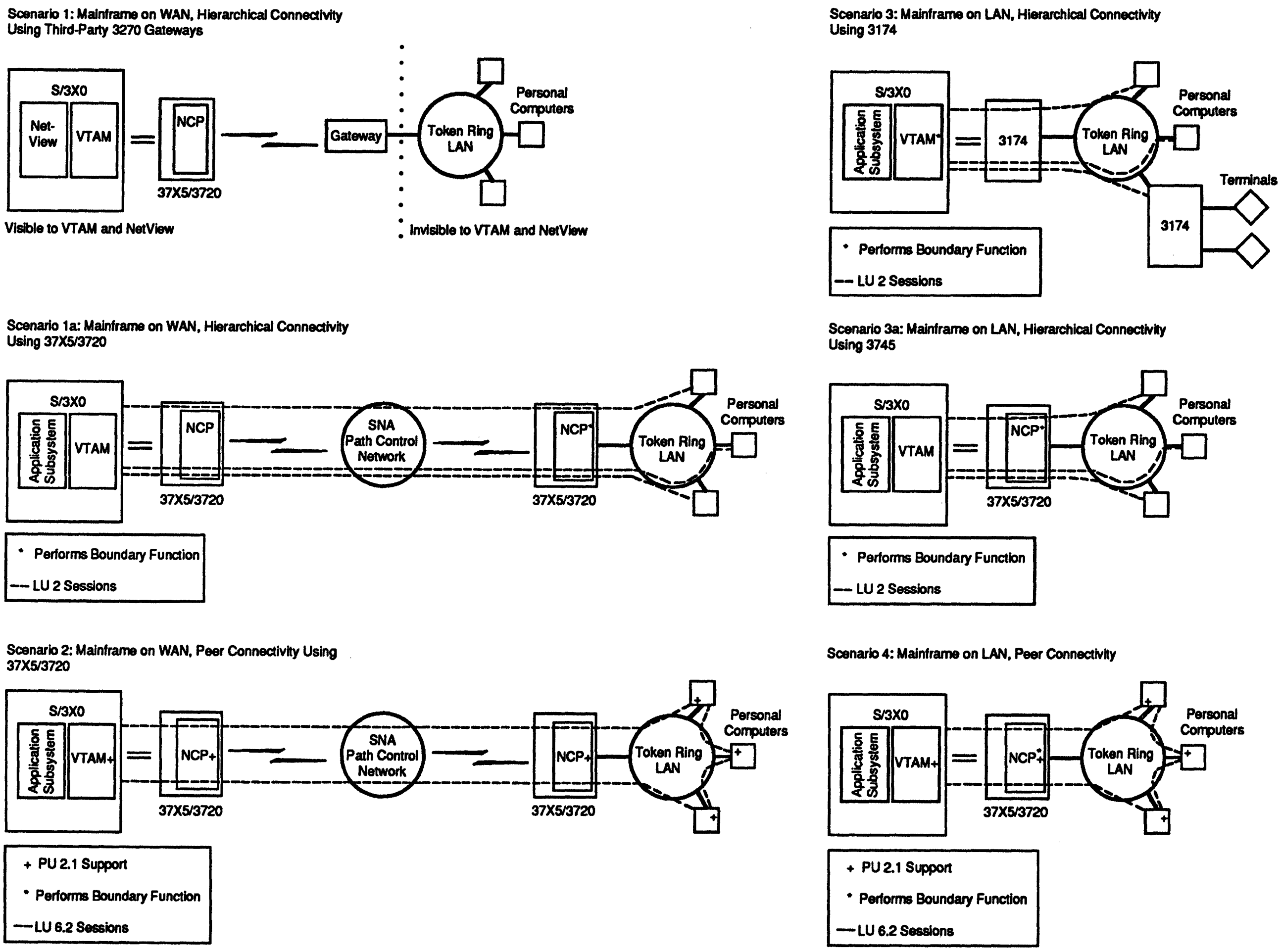
Scenario 3a: Mainframe on LAN, Hierarchical Connectivity Using 3745
A channel-attached front-end processor is another alternative solution if you have a locally attached mainframe and hierarchical sessions. This option is good if you have a need for high throughput and want to take advantage of the management capabilities offered by NCP, which runs only in 3745-class controllers.

Scenario 4: Mainframe on LAN, Peer Connectivity
In this scenario, the mainframe is directly attached to the LAN via a channel-attached enterprise controller and the appropriate releases of Advanced Communications Function/NCP and ACF/VTAM are used to support independent LUs and hence peer connectivity among end users. This option can yield the fastest response times and keep the mainframe from having to execute the boundary function conversion of SNA packet identifiers. The important distinction between scenario 3a and scenario 4 is the requirement for peer connectivity, not merely LU 2 master/slave connectivity.

Mainframe and LANs Collide

The mainframe and LAN worlds are no longer parallel universes. The two are increasingly converging as users seek to have their mainframes function as enormous file servers. Network designers and systems integrators looking to forge a LAN-to-mainframe link traditionally turned to SNA gateways that allow the mainframe's SNA network to interoperate with LANs based on XNS, TCP/IP, and other

Figure 3.
Typical Configurations for Connecting a LAN to a Mainframe



These scenarios depict the typical configurations of connecting a LAN to a mainframe depending o whether the host is local or remote and whether you want hierarchical or peer connections.

protocols. Gateways provide a quick fix, but present problems in the form of limited network management and throughput capabilities and lack of reliability.

Running SNA natively on the LAN can solve these problems, but at the price of complexity. SNA is no snap to install or maintain. There are architectural as well as technical factors to be aware of when deciding whether native SNA is right for your installation. The most important architectural factor is whether you will require independent or dependent LUs. The most important technical factor is the distance between host and LAN and whether local channel attachment is possible or remote attachment is required. Using IBM's new fiber optic channels or channel extenders will let you make a local attachment over campus distances.

Your requirements should drive the type of connection you make. If you expect to implement client-server computing, plan on implementing LU 6.2 (as described in scenarios 2 and 4) to enable peer connections. If you plan to have your personal computers work as 3270 terminals accessing host applications via LU 2, you have greater flexibility in your choice of controllers and configurations (as outlined in scenarios 1, 1a, 3, and 3a). The distance you need to traverse will mandate use of a local or remote connection.

Don't be intimidated by what may seem like an overwhelming amount of jargon, details, or choices. SNA is less fearsome than it first seems, but it will likely remain a very complex system. Since SNA was designed to support networks of dozens of mainframes and thousands of users, some of its features will seem like overkill in the LAN arena. However, there are definite advantages to running SNA over LANs. ■

PC Links to UNIX Database Servers

In this report:

Datapro Summary

Connecting PCs to UNIX database servers may not be the most conventional database configuration, but there are many advantages to this type of arrangement. Portability, connectivity to dissimilar platforms, and widespread DBMS server support are just some of the benefits of linking LAN-based microcomputers to UNIX database servers.

As you set up a client-server database application you may be faced with the following dilemma: The best operating system for your database server is UNIX, but the best environment for the client piece of your application is PCs connected on a LAN. Can you have the best of both worlds? Can you enjoy the robustness and scalability of machines running a database server under UNIX without giving up the variety of file, print, and administrative services offered by the most popular PC LAN operating systems?

This report will explore a number of possible solutions to this problem. Among them: linking via simple TCP/IP connectivity with UNIX as your (limited) network operating system; adding the TCP/IP-based network layer known as Network File System (NFS); the possibility of running a UNIX database server in addition to a NetWare, LAN Manager, or VINES server on your network (or how to run TCP/IP *and* another network protocol in a single PC); the UNIX versions of NetWare and LAN Manager; and an all-UNIX solution that incorporates DOS compatibility, SCO's Open Desktop.

This Datapro report is a reprint of "PC Links to UNIX RDBMSs" by Kevin Strehlo, pp. 50-57, and 74 from *DBMS*, Volume 4, Number 5, May 1991. Copyright © 1991 by M&T Publishing, Inc. Reprinted with permission.

Going through these gyrations doesn't make much sense unless you believe that UNIX has certain advantages over NetWare, LAN Manager, and VINES for supporting robust RDBMS products such as SQL Server and Oracle Server. Rather than make you accept that by fiat, we'll try to convince you.

Benefits of UNIX

UNIX is uniquely suited to be the operating system for a database server for a number of reasons.

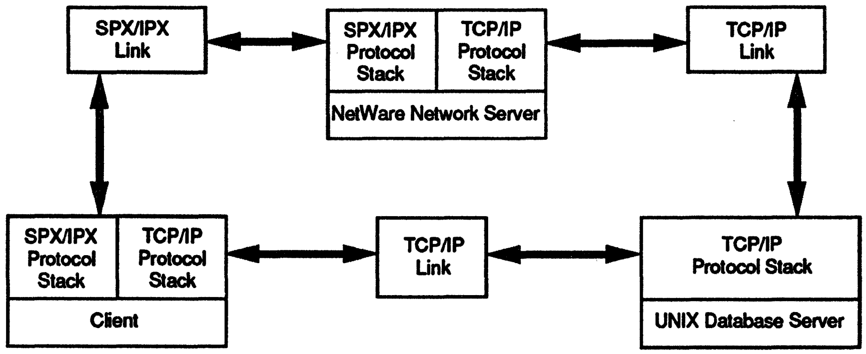
Scalability

For one, UNIX was designed from the start to be easily ported from one kind of hardware to another. As a result, a greater variety of computers—from a standard 286-based PC to a powerful supercomputer—run UNIX than any other operating system. This fact delivers a couple of very important benefits. The first is scalability. If you outgrow the capacity of the UNIX box that you're using for a database server, you can always replace it with a bigger, more powerful machine. In most cases, the changeover is trivially simple.

You can see the scalability UNIX offers without even leaving Intel-architecture computers. OS/2 is the most common operating system for Intel platform database servers today. But OS/2 cannot make use of more than 16 MB of RAM, while database servers can put more RAM to very good use indeed if the operating system doesn't get in

Figure 1.
The Worst Scenario

The worst scenario when linking to NetWare file and print services while simultaneously connected to a UNIX database server: two separate NICs in the client and two protocol stacks stealing precious DOS memory.



the way. A Tricord or a Compaq System Pro would be limited to a dozen or so Oracle users under OS/2 primarily because of memory constraints, for example, while the ability to equip such machines with as much as 256 MB of RAM under UNIX would greatly increase their Oracle capacity.

Moreover, OS/2 can make only limited use of a second Intel processor. Implementations of UNIX with symmetric multiprocessing can make good use of many processors—thus the Wyse Series 9000i, which scales up to as many as eight 486 processors, and the 10-processor version of the Zenith 1000, among others.

If you still aren't convinced, what about Microsoft? The vendor behind OS/2 LAN Manager is always saying that one of SQL Server's key benefits is scalability to the Sybase version of SQL Server on a larger UNIX platform. If you put up an application on a LAN Manager network and ran out of gas, you—or so Microsoft's story went—could simply replace your Intel-architecture machine running Microsoft's 16-bit version of SQL Server with the 32-bit Sybase version of the software running on a robust UNIX machine.

Interestingly, until the release of the Sybase PC Net-Library for SQL Server in late February 1991, that much touted scalability was more hype than fact, as we'll show.

Connectivity

A second benefit of UNIX is connectivity. Because UNIX runs on more computers than any other operating system, and because UNIX-based database servers can easily communicate with other UNIX machines, it becomes easy to operate on data distributed over dissimilar platforms. In addition, UNIX is a mature operating system with a wide variety of gateway products for connecting to other environments, including IBM's SNA.

Freedom

A third benefit of UNIX is that nearly every major database server product you care to name—SQL Server, Oracle, Ingres, Informix, and SQLBase, as well as lesser-known but interesting server products including Empress, Unify 2000, Progress, and Interbase—is available in a very mature version under UNIX.

Finally, you may want the database server on your LAN to be an existing UNIX host because that's where the data is already, sitting in a SQL RDBMS. By allowing PC workstations with advanced client software to access that existing data, you can enhance the productivity of the members of your organization who have to work with the information.

What's the Problem?

Unfortunately, each of the major network operating systems for PCs has its own preferred method of communicating among nodes on the network. Each method is embodied in a protocol that defines the form of data packets that are passed from one network node to another. For NetWare, the preferred protocol is SPX/IPX which is highly optimized for Novell networks. LAN Manager transmits information via NET BEUI or named pipes. VINES nodes connect to each other through a variety of protocols including Banyan's proprietary VINES transport layer protocols. In contrast to the protocols used by PC networks, UNIX machines typically communicate with each other using the TCP/IP, protocol. To make such a machine a full partner in an existing PC network, either the network must support TCP/IP the UNIX machine must support one of the network protocols, or the client must support multiple protocol stacks.

Figure 1 is a schematic diagram of a client attached simultaneously to a typical Novell server running NetWare 3.11 and to a UNIX database server via a TCP/IP link. In this particular configuration, the Network Interface Card (NIC) supporting SPX/IPX on the client cannot simultaneously support TCP/IP, so a second NIC is needed to communicate with the database server. The LAN shell driver and both protocol stacks must reside in memory simultaneously.

Given that TPC/IP software alone can sometimes occupy more than 110K, such a configuration has difficulty supporting an application of any substantial size in a typical 640K DOS machine. Even software like Quarterdeck's QEMM that can move drivers up into high memory on 80386 machines are not a satisfactory answer with two protocol stacks loaded simultaneously. Nor is getting the client running with *two* NICs offering potential base I/O address and interrupt conflicts easy. Nor is it desirable to have to spring for two NICs per client.

Before delving into alternative solutions to making multiple protocols available, let's examine what you can do with TCP/IP alone. If a network is being established for the purpose of running a new database application, the combination of TCP/IP connectivity and UNIX's built-in multiuser communications and security may be enough to satisfy your needs.

TCP/IP

TCP/IP (transmission control protocol/internetwork protocol) was developed long ago as part of the U.S. Defense Department's ARPA network. TCP/IP's popularity among UNIX installations is due in part to its being in the public domain, to its wide distribution as part of the popular (and free for the price of the media) Berkeley extensions to UNIX, and because it's a mature, solid protocol.

As the slash implies, the protocol breaks into two basic parts, TCP and IP.

IP is a way of linking subnetworks. In the OSI (Open Systems Interface) reference model view of the world (Figure 2), IP is the network layer that takes a packet from one data link, consults a routing table, and forwards it to another data link. That means each node must have a unique address, which in the IP scheme is a four part hierarchy. In essence, IP modules know how to get a packet to other networks via servers acting as routers (or via standalone routers). The router connects two TCP/IP subnetworks.

By itself, IP is of limited use. The internetwork protocol has no way of guaranteeing a packet gets delivered. It is up to the TCP transport layer to guarantee that packets submitted at one node will be delivered to the appropriate node error-free and in the order they were sent.

In the case of an Oracle client-server database application, the entity that makes direct use of TCP/IP is the SQL*Net portion of Oracle. SQL*Net maps the client's Oracle-specific protocol into a network-specific call, which is passed to the TCP/IP protocol call-handler.

Other services built on top of TCP/IP include Telnet (which implements the UNIX remote terminal logon), file transfer protocol (FTP), and SMTP (Simple Mail Transfer Protocol).

Of course, there are other aspects of a network operating system that TCP/IP and its utilities alone do not handle: security and network administration, for example. Yet a client logged onto a UNIX host through TCP/IP doesn't need anything more in that regard because it automatically comes under the control of the security and administrative tools by which a UNIX system administrator manages multiple users logged in via asynchronous terminals.

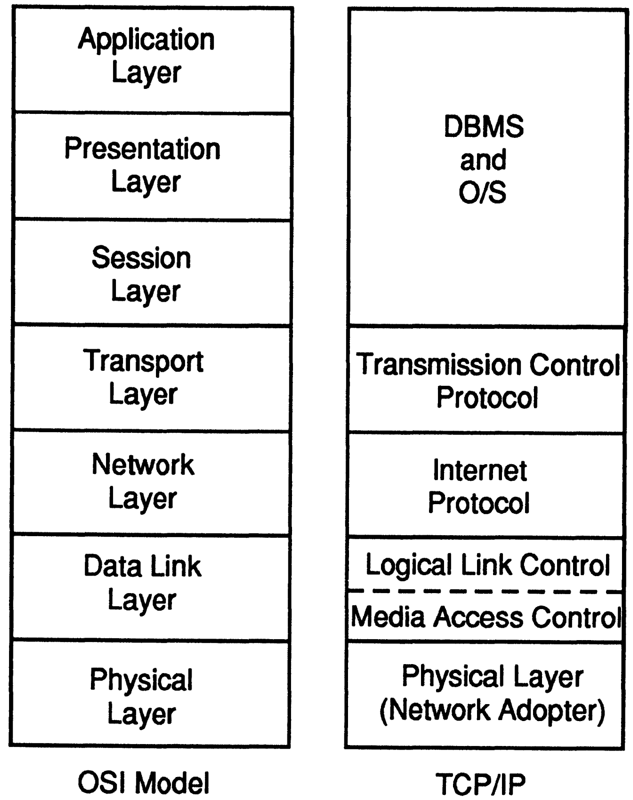
If your UNIX system is set up as a "trusted" system, for example, all the connected clients must obey the security policy of the system. The security policy is a set of rules that oversee and guard interactions between programs running on the system and files, devices, and interprocess communication objects. A trusted UNIX system provides accountability by associating each account with a user (in this case, a user operating an application on a PC client) and auditing every action. UNIX DAC (Discretionary Access Control) determines whether a user has access to the information requested.

UNIX system security and administration is extensive and far beyond the scope of this report; suffice it to say that all those protections and tools apply to PC clients just as they do to asynchronously connected terminals.

Some of the major Network Operating Systems also use TCP/IP as a method of communication between subnetworks. For example, it is possible to connect one NetWare network to another NetWare network in another city using a TCP/IP backbone. This technique, called tunneling, encapsulates a SPX/IPX data packet within a TCP/IP packet. They physical connection can be anything from an Ethernet to a satellite link that supports TCP/IP. At the receiving end, the TCP/IP shell is stripped off, revealing the original SPX/IPX packet, which is then routed to the target NetWare network and sent to the PC workstation that is its ultimate destination (see Figure 3).

And yet even with all the UNIX communications facilities, security features, and administrative tools plus its potential use for internetworking, TCP/IP is not the match of today's network operating systems. While it can provide

Figure 2.
The ISO 7-Layer Model



The ISO 7-layer model is the standard frame of reference for networking protocols even if ISO-compliant products are not rapidly displacing older solutions such as TCP/IP, RPC, and NFS. Layer 1 is where raw bits are transmitted over a physical medium. Layer 2 changes packets of data into frames and vice versa. Layer 3 routes data between adjacent network devices in the form of packets. Layer 4 passes data from the network layer and redistributes it the appropriate upper layers. Layer 5 negotiates and manages communication sessions between processes on the network. Layer 6, the presentation layer, has nothing to do with GUI: it performs application services such as data transformations, encryption, and compression. Finally, Layer 7, the application layer, provides file sharing, network transparency, and distributed processing.

the API-level connectivity required for a client-server application, it cannot provide some of the standard network services such as file serving and printer serving. One of the benefits of giving people a PC on a LAN rather than an asynchronous terminal, after all, is to enable users to run standard PC applications and take advantage of shared disk space and printers.

To make TCP/IP support a full suite of LAN-based services, Sun Microsystems developed two services that run on top of TCP/IP: RPC (remote procedure call) and XDR (external data representation). On top of that layer, Sun built Network File System. NFS, which has been widely licensed, allows a file system on one computer to appear as if it were a local file system on another computer, even if the computers are running different operating systems. In our frame of reference, NFS is used mostly to allow large UNIX machines to provide file services to networks of

What's NSF?

The network revolution has touched more than just the DOS-based user. UNIX connectivity has advanced to a level at which it allows several hundred UNIX systems to be networked, allowing users to share database files, printers, and CPUs.

To allow sharing of information across unlike network systems, the UNIX world came up with a network solution that operates on several hardware and operating system platforms—an independent network service designed to be operating-system and hardware independent. This communications solution is the Network File System (NFS) network.

Moving Outward from Sun

The Network File System (NFS) was developed by Sun Microsystems in 1984. Since then, Sun has licensed NFS to operate on more than 110 separate platforms, including most UNIX-based operating systems, such as SCO UNIX, Interactive System's 386/ix, and Digital Equipment Corp.'s Ultrix.

The NFS network allows the files systems of remote computers to appear to the local computer user as if the files were on his or her local drive. For example, if the user is required to read and write to a central DBMS, he or she would need only issue an NFS "mount" command and the DBMS file system would become accessible over the network as if it were local. Other users may also mount that file system to share information. In other words, NFS provides the same kind of file sharing as found in NetWare, LAN Manager and VINES.

NFS differs from the File Transfer Protocol (FTP) found implemented as part of most TCP/IP environments because FTP does strictly what the name implies—the remote file is transferred to the local system before processing can begin. NFS makes the file appear to be local while it remains physically on the remote server.

NFS is designed to be machine and operating system independent so that any computer system can supply files to many different computer platforms. It is not limited to the UNIX world. Other operating systems supported include MS-DOS, Macintosh, VAX/VMS, and IBM's mainframe operating systems VM and MVS.

To provide operating system independent network functionality, NFS uses a technique known as Remote Procedure Call (RPC). RPC is much like a local procedure call in its behavior and utilization. With RPC, a local routine can call and bind to a remote service. RPC uses another set of routines called the External Data Representation (XDR) library to account for differences in the internal data of different hardware platforms. XDR represents a standard, intermediary data representation that all internal data representations can be translated into quickly. Basically, then, the NFS protocol is a library of remote procedures that may be used to connect to remote files, translate those files to XDR format for transmission via TCP/IP, and convert them back to the local file format on the fly.

A server doesn't send directories over the network as part of the exporting process; instead, it advertises in a control file. These files provide information on available directories and their access parameters.

Once an NFS link is established, a program running on a local client can issue conventional I/O calls to the remote server's files. Physical location is essentially transparent; commands such as READ, WRITE, and SEEK work as they normally do.

NFS and the OSI Model

As its name denotes, the International Standards Organization (ISO) is an international organization that creates standards. ISO has established a standard communications model called the Open Systems Interconnection or OSI, a seven-layer model in which each layer is a type of protocol and addressing method. Its purpose is to provide network developers with a set of standards with which to provide the user community compatible networking products made by different vendors.

While many experts predict that OSI products will someday dominate the network world, NFS and the underlying TCP/IP, XDR, and RPC software on which they are based were well-established in computer sites around the world long before the first OSI-compliant products shipped. NFS and its support layers are unlikely to go away anytime soon, even though they don't support the much-touted standard.

PCs. However, Novell's recent announcement of NFS support as part of NetWare 3.11 is intended to allow UNIX workstations to access NetWare services. NFS is described in more detail in the sidebar "What's NFS?"

More Than One Protocol

Earlier I promised to delve into the intricacies of mixing protocols on your network wire—one protocol to link to the network operating system server present on a typical PC LAN and the TCP/IP protocol to link to the UNIX database server.

I implied, but didn't actually say, that it was impossible to connect to one of the big three network operating systems *and* to a UNIX host on the same network without mixing protocols. I was fibbing.

Banyan VINES allows the use of several protocols at what correspond to the network layer of the OSI model: TCP, UDP, and IP are among them. Thus, a TCP/IP protocol can link a single PC client to both the VINES server and a UNIX host.

VINES is far and away in the minority, however. Industry analysts say NetWare runs on anywhere from 50% to 70% of PC networks. And while LAN Manager is implemented on a much smaller percentage of LANs, it is a frequent choice for companies implementing client-server database applications. Thus the demand for solutions that support multiple protocols.

Multiple Solutions

Way back in 1988, BICC Data Networks provided the ability for PCs attached through TCP/IP to a UNIX host to also support other network protocols. BICC's ISOLINK

Nevertheless, the OSI model (see Figure 2) is a useful way to understand the networking world represented by TCP/IP and NFS. In most cases the physical layer is either Ethernet (IEEE 802.3) or Token-Ring (IEEE 802.4). The link layer protocol is Ethernet or Logical Link Control. The network protocol is the Internet Protocol (IP). The Transport protocol used by NFS is the User Datagram Protocol (UDP). NFS doesn't need a Sessions layer function. At the Presentation layer, NFS makes use of External Data Representation (XDR).

Installing and Operating NFS
An NFS network requires a networking interface card in the remote as well as local computers. IEEE 802.3 (Ethernet) is the most popular platform for NFS; it dominates the UNIX world and operates on a variety of hardware platforms. Most minicomputers come with an Ethernet.

Installation methods for the NFS network vary due to the number of different vendors that provide the SUN licensed software, but none are inherently complex. On the UNIX side, software installation for *all* software is standardized in a relatively simple and well-automated fashion. Install one package and you know how to install them all.

If you have a problem, it will likely be with the Network Interface Card drivers. Most Network Interface Card vendors provide utilities to test the cards, mostly for DOS. It's not a bad idea to use these DOS test programs before booting UNIX, because UNIX is relatively intolerant to hardware problems.

Discussion of some of the commands UNIX provides for checking connectivity, including Ping and Netstat, is included in the main body of this report.

When the installation is complete, consult your user manual to set permissions and other configurable features. Consult the NFS network manual to find out the precise remote file system mount procedures, which differ from operating system to operating system. In many cases you will decide to put the NFS commands in the AUTOEXEC.BAT or the UNIX mount tables.

LAN/UNIX system administrators may need to create mount points on local computers for file systems to be mounted from a remote machine. The mount point is the directory in which the UNIX or DOS file system places the remote mount file system directory.

Sun's Implementation
Sun's Release 3.01 of PC-NFS goes a bit beyond the original NFS by adding file locking support for the network versions of Clipper, dBASE, FoxBASE, and FoxPro. Print redirector and spooler support includes PostScript.

You'll be pleasantly surprised to learn that Sun's NFS redirector and TCP/IP stack only about 80K on a DOS client. Network cards supported include the 3Com network adapters, the Ungermann-Bass NIU, and Western Digital's WD8003E adapter.

Installation on the PC client is simple—the only detailed information required is the name and Internet address of the NFS-equipped machine(s) acting as a server. A supplied NET USE batch file links a designated logical drive to disk volume on the NFS server.

Sun's PC-NFS has a list price of $395 per node. The underlying TCP/IP software and several utilities including FTP are included.

NFS on the UNIX Server
The NFS service on a UNIX machine is carried out by deamons that read the configuration files set by the administrator. Deamons are programs that operate in the background mode of the UNIX operating systems. The number of deamons launched to control the NFS network is configurable. The administrator must find a happy medium—to many deamons can kill performance on the CPU, and not enough will kill the network.

An NFS network can be reconfigured at any time by the system administrator. File systems may be unexported or exported, permissions changed, hardware altered, and mount tables altered. Consult the NFS network guide before undertaking any of these processes.

NFS, like all network services, may fail at the server, client, or network level. NFS comes with several methods to determine and diagnose trouble on the NFS network. Coming from a Novell or 3Com world, you may find these diagnostic programs primitive and difficult to use. Welcome to UNIX.

—By David Linthicum

Multiprotocol Support software allowed several protocol stacks to be loaded into memory, and users could hot-key between them instantly.

Other vendor-specific approaches to the same problem have surfaced since. For instance, FTP Software offers the Packet Driver for Ethernet, which enables network cards supported by FTP's TPC/IP suite to run multiple protocols. In effect, the Packet Driver acts as a traffic cop, directing each packet to the appropriate protocol stack.

Gateway Communications' LAN adapters can be configured to operate NetWare and TCP/IP protocol stacks concurrently, eliminating the need for users to reboot to move from one protocol to the other.

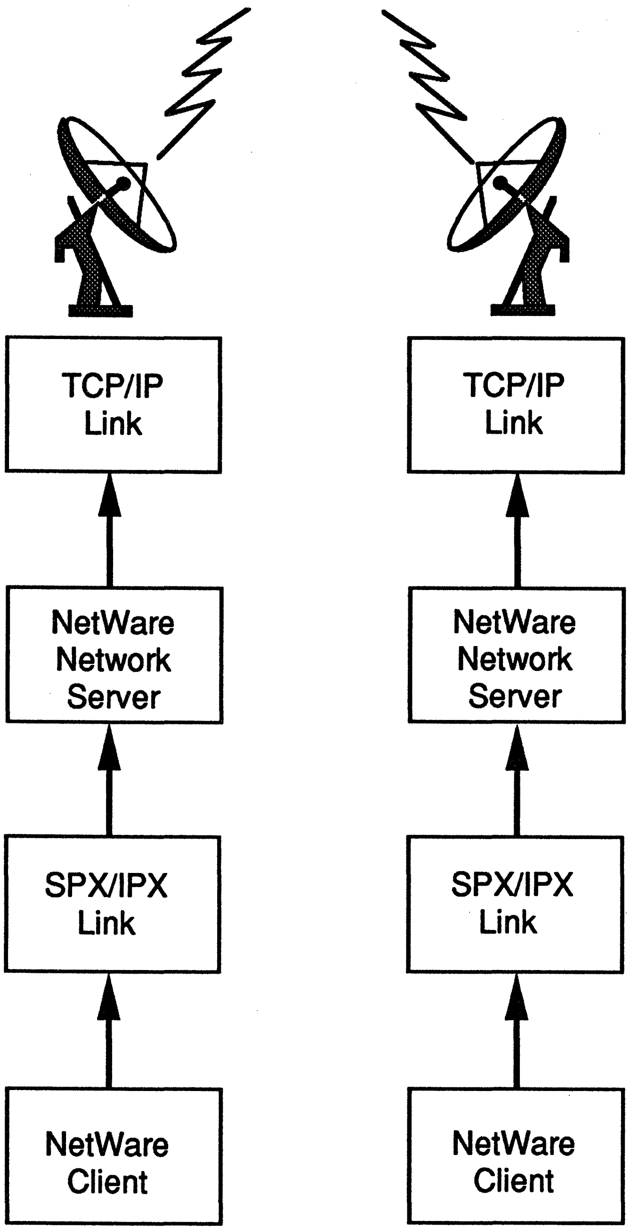
An innovative new way to avoid the RAM cram problem caused by having to support multiple protocol stacks is embodied in Open Windows, a product of Cogent Data Technologies, Inc. of Friday Harbor, Wash. Open Windows is a multiprotocol manager designed to be loaded in high memory to support a Windows 3.0 client. None of the lower 640K is taken up with drivers or stacks. Full Windows functionality is maintained, allowing the users to cut data from a UNIX window and paste it into a NetWare or LAN Manager window.

NDIS, ODI's Generic Approach

Today, two competing approaches championed by Microsoft and Novell—NDIS and ODI, respectively—offer a more generic ability to run several different network protocols at once almost regardless of which vendor's NIC is being used.

Most connections to NetWare networks involve loading an IPX layer. The IPX links intimately to your network adapter card; in fact, it must be "built" specifically

Figure 3.
NetWare v3.11



NetWare v3.11 can encapsulate SPX/IPX packets inside TCP/IP packets and ship them across a campus via an Ethernet backbone or across the country via a WAN connection.

for your particular type of card by your database administrator. On top of that, the NET3 (or NET4) redirector layer routes DOS file calls across the network to the NetWare file server.

ODI (Open Datalink Interface) replaces the monolithic IPX stack with a hardware independent layer called LSL (Link Support Layer) that must be written by the card vendor, and on top of that; and Novell drivers that link the LSL layer to any of a number of network protocols. All card manufacturers must do is write an interface layer that links their card to the LSL API, which Novell has pledged to maintain without change. Then Novell's driver to link ODI to that protocol takes care of the rest. Under the previous scheme, in contrast, 3Com had to write a monolithic hardware-to-protocol driver for each protocol.

For example, to use one of 3Com's Ethernet cards, you begin by loading the LSL driver 3C503 into memory. Then, if you need to communicate with the server via the IPX protocol, you load Novell's IPX-ODI driver after 3C503. If you need to handle TCP/IP, load Novell's TCP/IP driver. If you need both, you can load both.

One nice thing about Novell's ODI (Open Datalink Interface) is that it is possible to load and unload any part of ODI at will. Thus, you can maintain a constant IPX connection to a NetWare server and load TCP/IP only when it's needed for a connection to a UNIX database server. Or you could unload IPX layer first to make more room for your TCP/IP link and database client application.

Thus far, only a limited number of NIC manufacturers have shipped the necessary LSL layer, however.

The Microsoft NDIS approach, which is similar in architecture and advantages and has also been adopted by Banyan for VINES, has been available to board vendors longer and has therefore garnered more extensive support.

Sybase Catches Up

NDIS support for multiple protocols alone does not solve the problem of connecting PCs on a LAN to Sybase SQL Server running on platforms other than OS/2. Whereas Oracle had various versions of its SQL*Net layer to map the client workstations Oracle-specific calls into a particular network protocol layer call, Sybase was empty handed.

The February release of Sybase's PC-Network library finally provides the network-independent support for PCs that SQL Server had been lacking. Protocols supported by the PC Net Library interfaces included TCP/IP, DECnet, Microsoft LAN Manager, Novell NetWare, and AT&T StarGroup. Thus, it becomes possible for DOS, Windows, and OS/2 client applications written to a single SQL Server DB-Library API to communicate with SQL Server running on not only OS/2 but also UNIX, VMS, and MVS. One long-standing gap between Oracle and SQL Server has thus been narrowed. Just as you can with Oracle's interchangeable servers running on different hardware, it is now possible to replace your OS/2 machine running SQL Server with a VAX, a Digital RISC/Ultrix machine, an HP 9000, an IBM RS/6000, a Sun SPARCServer, a Sequent Symmetry machine, or a Pyramid MIServer.

Portable NetWare, LAN Manager/X

To allow their machines to function as NetWare servers, many minicomputer vendors have announced Portable NetWare for their machines.

Portable NetWare is a source-code version of NetWare 386 that allows third-party vendors to port the environment's API on top of the UNIX operating system that runs on their hardware. In effect, the minicomputer becomes a NetWare server providing file and print services while also supporting UNIX applications including RDBMS server products.

For example, Interactive Systems Corporation of Santa Monica, Calif., sells a version of Portable NetWare compatible with its UNIX operating system. Interactive's Portable NetWare runs on any 100% PC-compatible 386 machine. A similar product developed jointly by Microsoft, Hewlett-Packard, and 3Com, called LAN Manager/X, is available from a number of vendors including Santa Cruz

Operation, DEC, H-P, AT&T, and Unisys. In the same way that LAN Manager sits on top of OS/2, LAN Manager/X sits on top of UNIX.

The fact that you don't have to support TCP/IP alongside the native network protocol may not be the memory saver you'd expect in these Portable NetWare and LAN Manager/X implementations, however. One example is AT&T's StarGroup package. Clients can be DOS, OS/2, or UNIX workstations running NetBIOS, AT&T's UNIX Transport Interface, or named pipes, all supported on top of an OSI-compliant transport. Unfortunately, support for all those transports and interfaces leaves only about 420K left for applications on a DOS client that lacks extended memory to load drivers high.

Or Go Native

One last solution if you need to connect PCs to a UNIX database server is to go native—run UNIX on the clients as well as on the server. That means the clients should be fairly robust machines, probably 386s. One attractive UNIX solution for that kind of robust client environment is SCO's Open Desktop software.

Open Desktop incorporates all the services needed to put together client-server database applications without giving up DOS compatibility. Based on UNIX System V, Open Desktop includes networking via a TCP/IP suite, support for MS-DOS applications running within a window, and SQL database support.

That SQL support comes via a Database Services module that incorporates a version of the Ingres version 6.2 RDBMS (although other SQL databases including Informix, Oracle, and Sybase can be used instead). The Open Desktop workstation packages includes query-by-form and report-by-forms facilities for Ingres.

For lower-level front-end development, the Open Desktop developer's package includes an embedded SQL preprocessor for C, while preprocessors for FORTRAN and COBOL are available from Ask/Ingres. Other Ingres front ends available for the environment include Ingres Windows 4GL. Applications developed using this Ingres tool kit on Open Desktop can be moved transparently to other workstation-based graphic platforms as well.

For connection to data on platforms other than UNIX, Open Desktop leverages the many Ingres gateways. Ingres gateways exist for many RDBMS (and flat file system) products running on most big iron platforms.

Finally, Open Desktop enables the creation of advanced graphic user interfaces on relatively low-end PCs or X Terminals via built-in X Windows support; X Windows is a multitasking windowing and graphics system that applies the client-server model to graphic user interface. Under the X Windows model, each workstation has its own X Display server process running on the network. That server controls the screen, keyboard and pointing device for one or more workstations. The developer simply links the client program with Xlib, a library of graphics and windowing functions that translate the graphics operations in the X communications protocol for display dictated by the Server. ■

LAN Applications

In this report:

Datapro Summary

Installing a local area network can be a challenging enough process, but now comes the time to put your multiuser software applications to work. The most widely used network applications are database management systems, word processors, and spreadsheets. It is important to recognize the differences between single- and multiuser versions of software applications before they are executed. Issues such as installation, data integrity, concurrency, and performance in multiuser applications are explored.

Introduction

Will the applications you bought the computer for work on your LAN? Since networks turn virtually all applications into groupware, the first thing you need to determine is whether or not the application is appropriate for group use. There are two criteria you should apply to this decision.

First, not all software is meant to be used by more than one person. Many utilities come under that single-user category. *Utilities* are programs that individual users apply to enhance their computing. Second, single-user versions of any program may not be shared among users on a network. In the first place, such sharing usually violates the developer's license. Also, in most cases, multiple users attempting to use a single-user program end up corrupting data and getting in each others' ways. The bottom line is to use single-user programs as a single user and get multiuser versions of any program you plan to run on the network.

Let's go back to that comment about groupware, a current buzzword. Sometimes, with a lot of vendor hype flying about, it's difficult to nail down exactly what the term means. Often it's applied to products that are, in reality, nothing more than menu shells with electronic mail facilities and a few nice utilities. The utilities, usually some form of calendaring and group scheduling, are what lead the developers to call their products groupware. In a sense, that's probably true. However, the real fact is that any program workers can use as a group to improve their efficiency could fairly be called groupware. That, if you think about it a bit, includes virtually all multiuser software operating on a LAN.

If you think of groupware in those terms, you come up with another challenge. You need to install your groupware on the LAN so that it's of maximum use to the group. *Maximum use* means that all members of the group who need access to the program and its related data can easily get that access. All those who don't need access are shielded from the software.

Why "shielded"? Simply because most security breaches that result in damage to or loss of network data come from unintentional errors caused by untrained or novice users. The largest group of those users includes workers without the need to use the program. Since they don't need to use the program, they have limited, if any, familiarity with it. Should they find themselves lost in an unfamiliar application, there's a high probability of damage resulting from efforts to escape. In other words, efficient

This Datapro report is a reprint of Chapter 7, "Will My Applications Work on a LAN?" pp. 117-137, from *Executive Guide to Local Area Networks* by Wayne Rash, Jr., and Peter R. Stephenson. Copyright © 1990 by *COMPUTE!* Publications, Inc. Reprinted with permission.

groupware is there for those who need it and invisible to those who do not.

In keeping with that philosophy, we'll spend the rest of this report talking in general terms about how you install multiuser applications and look out for problems. We'll use a few general examples and be pretty vague about the particulars of individual networks. However, this report is specific about the method of drive and directory layout, so you can apply some general principles to your particular network. With very few exceptions, all LANs organize the same way. They each have different commands and terminology, but the general layout of network drives and directories is pretty similar. Use this report in conjunction with your network administrator's manual and the installation guides for the application you're installing.

We'll cover three basic types of applications. These are database management systems (DBMS), word processors, and spreadsheets. Then we'll discuss some other applications such as e-mail and menu shells. Finally, we'll touch briefly on some typical LAN services that often are available. In addition to user and administrator utilities, many LANs have their own e-mail and menus. We'll talk a bit about the differences between network supplied versions and their third-party counterparts. Let's start with a discussion of the issues that surround multiuser systems. Let's begin with database management systems.

The DBMS: Sharing Data in an Orderly Manner

When one user accesses a database on a single-user system, the only consideration is getting at the data. When more than one person uses a DBMS on a LAN, several other potential problems surface. In a single-user system, the ability to ensure data integrity is, although not trivial, a fairly straightforward issue. On a multiuser system, data integrity is constantly at risk. On a single-user system, there's really only one way to handle application and data files. On a LAN there are several. If you're getting the idea that a DBMS on a LAN requires users and administrators to use a bit of extra care, you're exactly right. Let's draw a few specific comparisons between single and multiuser DBMS.

Data Integrity

Let's start with data integrity. Without going into deep technical details, *data integrity* refers to the ability of database management to ensure that every record is unique. It means there should never be two copies of the same record, and that when the user searches for that record, only that one can be retrieved. In order to achieve this kind of integrity, in addition to having only one record, the record's location must be definable in very specific and unique terms. The definition of the location cannot point to more than one record.

In a single-user system, that's pretty easy to accomplish. But in a multiuser system where more than one user is accessing the same data, it's possible to cause duplicate records. This can occur when performing tasks known as relational operations. A *relational operation* is one that involves the use of more than one database or data table. An example of a relational operation might be combining the names, addresses, and phone numbers of companies in one database with their orders for supplies of some sort from another. When such a relation exists, it's necessary to have at least one piece of data (called a *field*) that is common to both sets of data *(databases)*. This piece of common data is called the *linking* or *related field*. If several users are performing relational operations using the same linking field, there's a danger of losing data integrity. A good multiuser DBMS will guard against that by not permitting changes to the linking field's contents without warning other users.

Concurrency

Concurrency control refers to the database's ability to prevent changes to any piece of data by more than one user without notifying the other users of the change. Here's an example of what can happen when you lose concurrency control in a database: Let's suppose two users are updating the same record. While one is updating, the other is *browsing* (reading the record only). After browsing awhile, the second user decides to update the record (write to it). However, the second user is unaware that the first user updated the record and so changes it again. Now the record has been changed twice and neither user realizes it. A good multiuser DBMS prevents this by forcing the second user to read the results of the earlier update prior to performing the second update.

There are other concurrency control issues, of course, but they spring from this one. Obviously, in a single-user system concurrency problems could never occur.

Performance

The third issue separating single and multiuser database systems is *performance*. On a single-user system, performance is largely a product of the performance of the application itself, the speed of the computer's processor, and the access time and condition of the computer's fixed drive. If the application is fairly efficient, the computer is an AT, and the hard drive is fast and optimized, you'll get reasonable performance. Degradation occurs as the data files get bigger and the use of relational operations increases.

However, on a local area network these issues are complicated by the addition of network performance and the number of users in the DBMS at the same time. Although it's unlikely that the network operating system can be slowed materially by the number of processes on it, there's a serious possibility of slowdown at the network drives as the number of attempted accesses increases. This can become very pronounced when a large number of users are accessing a particular type of network database program. To understand this problem, let's briefly review what we know about file servers.

A file server, you'll recall, is simply a substitute for the drive in your stand-alone PC. If your LAN is set up and tuned properly, you'll never know you're not using your local drive. All programs and data are loaded into the workstation's memory from the network drive exactly as if they were being loaded from the PC's local drive. No work is actually done by the file server.

With that in mind, if you have one user accessing a single drive on a local computer for his or her database management tasks, the drive is idle between accesses. If, on the other hand, several users are accessing a similar drive organized the same way, they're bound to get in each others' way from time to time. These collisions can occur when two users want the same record, when the drive is busy at the moment a user tries to access, or when an entire file is locked during a batch process such as indexing. The trick is to keep those collisions at a minimum. This can be done, but it's very difficult to do on a file server. Therefore it might be necessary to have a different database configuration called a *database server.* Sometimes called a *client server,* the database server system was designed specifically for the multiuser environment. It prevents (or at least limits) collisions of the kind just described by letting the server do some of the work. The work the server does is directed at the data and its management. It acts under instructions from the workstation (client).

In other words, the client processes the application, issues instructions for manipulation of data to the server, and processes the results of the server's work. The server manages the data and performs the data-related functions. It also keeps track of where the data resides (it can be in many locations), controls integrity and concurrency issues, and keeps track of individual transactions in a journal. The database server only goes out looking for data and writing to it in the final tiny bit of time in a database transaction. Before the transaction can take place, the server must receive an instruction called a *commit.* If anything happens that might corrupt the data, the operation is aborted, or rolled back. The data maintains its integrity, even though the system might have crashed in the middle of the transaction. In addition, since the actual write to the record lasts only a very short time, the record is locked only for an instant. This helps reduce performance degradation by limiting the amount of time the disk is in use.

So, there are two basic types of multiuser DBMS: the file server and the database server. Which is best for you? Database server systems are rather complex and expensive. File server applications are less complex and less expensive. As databases and LANs get larger, you'll need the database server. On small networks you can often get by with a file server DBMS.

dBASE III PLUS, dBASE IV, Paradox, and *R:BASE* are examples of file server databases. They are, essentially, single-user systems with some additions to prevent multiple users from colliding. Oracle, SQLBase, and SQL Server are examples of database server products. They're designed to use data located on several servers within a network (distributed processing) and to process transactions individually. We'll explore how each of these types of DBMS should be set up on your LAN.

The File Server: Simple to Install, Simple to Use, Moderately Reliable

The file server type of database is, essentially, the multiuser version of a single-user application. As such it has its drawbacks in terms of how it manages data in a multiuser environment. However, in a smaller system, the file server database has a few advantages. First, many smaller LANs spring from an office environment where there were standalone computers in wide use. Since this scenario implies an opportunity for networking, the next question is, quite reasonably, "How am I going to immigrate all my database applications onto the LAN?" In this case, it makes sense to simply switch to the multiuser version of the single-user DBMS in current use. Sometimes the applications you've written with the single-user product will need slight modification. But, in the main, you'll be able to use what you have without much pain during the conversion to the network.

The second advantage of the file server DBMS in a small LAN is that it's usually easy to install and administrate these less complex products. We keep emphasizing smaller systems because LANs with more than ten to fifteen users on a database tend to be very poor performers unless applications written for them are very carefully constructed and tuned. So, let's assume that you've decided the file server DBMS is for you. What do you do next? Let's examine the general configuration of a LAN for file server database management.

When the LAN administrator set up the LAN, several classes of directories and network drives were created. The exact configuration varies with network vendors, but in general here's the usual layout. Each user has his or her own directory, usually called the user's *home directory.* This directory contains data and programs that are for use by that user only. Other users are typically restricted from entry.

In addition, there are system directories, application directories, and data directories. These directories can be mapped or linked to network drives, but they're the usual divisions for most LANs. Into such a system the file server DBMS fits very well, indeed. The database management program itself (*dBASE III PLUS, dBASE IV, Paradox,* or other application) should reside in the applications area. In most instances, this will be a separate network drive with individual directories for separate programs.

If there are applications written for the database management system, they'll reside either in the applications area or in separate areas, depending upon their use. For example, if only one user needs a particular application and data, the network supervisor might choose to locate both in subdirectories under the user's home directory. Take care here, however. In most cases these applications need the actual database program in order to function. That means there must be a path to the program that the application can recognize. Likewise, the program must be able to find the application's data.

There's another way that, if the network permits it, is much superior. Actually, all applications belong in an application subdirectory near the database program, with the data in a similar subdirectory. The problem is that you might want to restrict other users from access to the data, the application, or both. If your LAN allows you to restrict users on a file level, all you need to do is password the file. Then only authorized users will be able to get at the data or application. This might require some minor modification of the application, but from an organizational perspective it's the right approach. If your LAN doesn't allow protection of individual files, you'll have to move the data and application into your home directory.

When several, but not all, users need access to a database application and data, there's yet another way to approach the problem of where to locate the files. You probably wouldn't want to use this approach for a single user since it does take up additional LAN drive resources. This method involves assigning a separate subdirectory to the application and data. The application is in the first level subdirectory in the applications area and the data is a subdirectory under the application. It's usually easy to provide password protection to directories and subdirectories. Be careful to ensure that the second level subdirectory, the one with the data, is accessible through the password for the application subdirectory.

So, let's review the configuration of the LAN for a file server database management system. The program goes into a directory on the applications drive. The data and applications go into separate subdirectories under the program. If there are applications and data that are restricted to just a few users, they might have their own first- and second-level subdirectories. If there are applications and data for just a single user, they can go in with other applications and data if your LAN allows password protection of individual files. If not, they can go into subdirectories in the user's home directory. As an option if your LAN allows password protection of individual files, all applications and data may go into subdirectories under the main program. Figure 1 diagrams what all this means.

Database Servers: Power and Flexibility . . . at a Price

If you're starting from scratch or if you have fairly large requirements, you'll want to look into a database server configuration. Sometimes called client-server technology, the database server offers the ability to address data wherever it resides, speed, and high performance from the network drives. The way such a system works is as follows.

Data can reside on several servers of various types. The workstations (clients) have some method of locating the data. The application resides either on the client or the file server. This means the application program can be on the client's local drive or on the LAN's file server; it's always executed from the client, of course. When a user executes an application from his or her workstation, the application reads a file called a *data dictionary*, which tells the client where to look for the data. The application then issues instructions to the server containing the data. The server executes the instructions and returns the result to the application on the client for further processing. This eliminates the need for the client to actually manipulate the data and try to keep track of all the individual transactions of the database or data table.

The efficiency comes since the actual data is never totally offloaded to the client. This saves access time to the network drive containing the database. When the database must actually be addressed, the action is very brief, again saving time at the drive and improving network and application performance. Finally, the database server keeps track of complete and incomplete transactions in a journal so that, in the event of a failure, they can be reproduced to ensure data integrity. The use of the data dictionary and other network refinements allows clients to have access to data wherever it resides without having to know specifically where it's located. This approach is called *distributed processing*.

Now, how do you install one of these beasts? Actually, in some respects, the configuration of the network is a bit simpler than for the file server system. However, in most cases, the installation of the program itself isn't for the faint-hearted. We won't go into that level of depth here because the various programs differ fairly significantly in their installations. But, as a start, here's how the network will be configured to receive this kind of system.

First, data can reside on any legitimate server. That means the server must be accessible to the client. This can either be direct access or access through a bridge, gateway, or some other method of connection recognized by the client's own LAN. For most client-server applications, it doesn't matter that data is located in multiple locations. How persnickety the system is in that respect depends upon the individual database product. Some products allow more or less multiplicity of data than others. Before you settle on a product, make sure it meets your data management requirements.

When you locate your data, be sure it's in its own directory or on its own network drive. Ensure that the drive or directory has password protection for the users who need the data. The applications reside on the file server appropriate to the individual clients (the one serving their LAN). Again, they should be password protected and in their own directory on the applications drive.

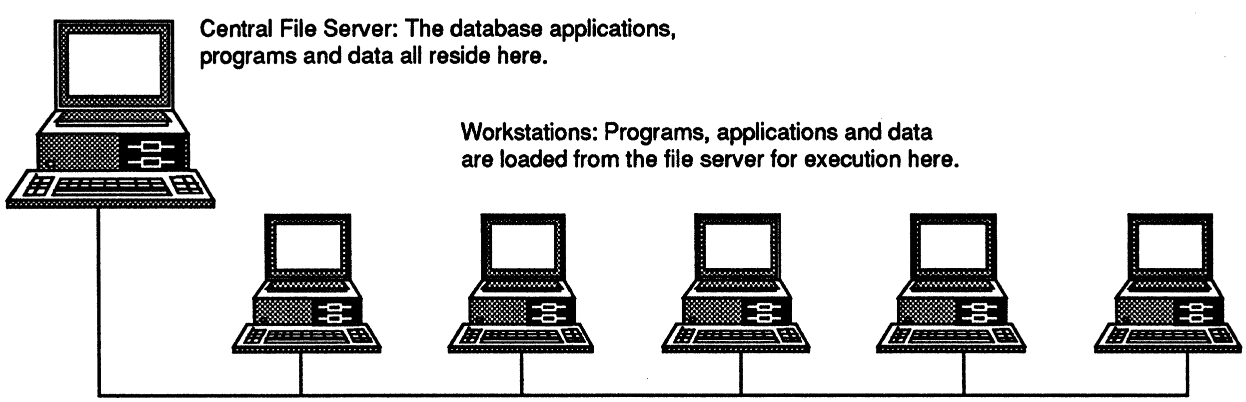
Here is where the two types of DBMS vary. In the file server DBMS, there's just one program with its attendant applications and just one set of data. In the database server system, the database management part of the system is a separate program from the application. The database management program resides on one or more database servers, while the application part of the program resides on the client's file server. It's really pretty simple in theory. In practice, the simplicity is greater or smaller depending upon which product you choose. Figure 2 illustrates a database server configuration.

Word Processing: A Bit of a Different Beast

Word processor programs for LANs can be as different from each other as night and day. Some keep track of the number of users, some require special user configuration files, and some allow customization on a user-by-user basis. Let's begin our discussion of word processing on a LAN with a few generalizations.

First, there are three basic locations for the files associated with word processing. The first is the applications

Figure 1.
A File Server Database Configuration



drive. The word processing program itself resides in its own directory on this drive. Users should have appropriate access to this directory.

The second word processing area is either a subdirectory under the user's home directory or a separate subdirectory under the application itself. This is where the documents prepared by the word processor users reside. If the documents require no access by users other than the creator, they can go into a subdirectory under the user's home directory. If they require access by several users, they go into a public subdirectory under the application's directory. If the documents need to be restricted, they can usually go into a private subdirectory under the application. Some word processors such as *WordPerfect* actually allow password protection of the individual document.

The third basic word processing location is for user configuration files, if your word processor uses them. These go into the roots of the user's home directory. They must always be in the boot path for the individual user or the word processor won't be able to find them. These files usually are required to allow access to the program and to provide customization for the user's word processing environment.

Installation

Installing a word processor is usually a two-step process. The first step involves loading the program and creating the appropriate directory areas such as those described above. The second step is the customization of profiles and authorization of users. Different systems have different ways of handling these issues. The rule of thumb is that the products with the most features also tend to be the most customizable in terms of user profile or configuration files. If you're in the process of selecting a word processor for network use, we recommend that you look most seriously at products that have full-featured network versions. Smaller programs that are advertised as "network aware" are usually nothing but trouble. Here's the reason.

Data Integrity

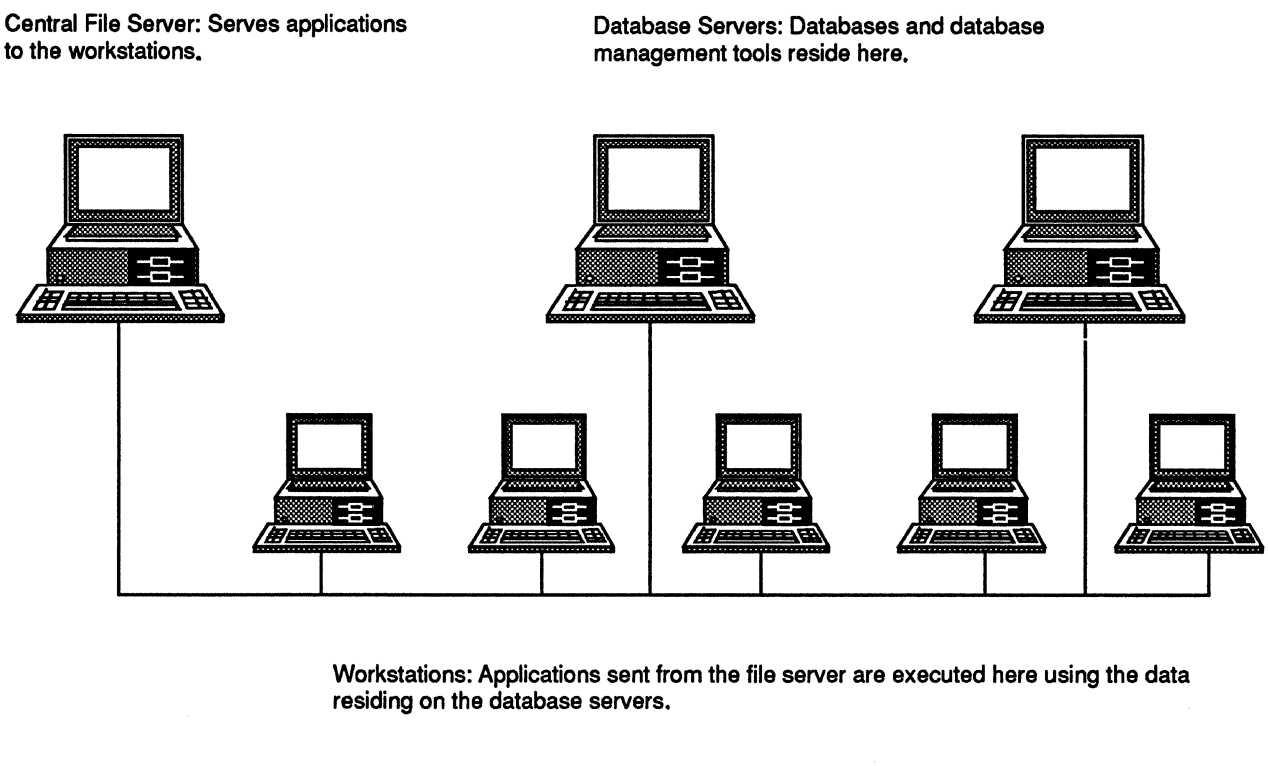
In a network environment, some of the aspects of data integrity and concurrency control we discussed in the section on databases hold true for word processors as well. Two important types of protection in word processors are needed. First, you need protection from keeping legitimate users out of the word processor simply because another worker is using it. Second, you need protection from corrupting or simultaneously updating documents when more than one user needs simultaneous access. That's our old friend concurrency control dressed up in word processor's clothing.

The solution to the first requirement is the assurance that multiple users (within the limits of the product's license) can access the program simultaneously over the network. The solution to the second is to ensure that if two users attempt to write to the same document at once, they're either prevented from or forced into saving their changes under a different filename. The second option is, of course, preferable. If your word processor can't handle both of these requirements in a manner that makes sense in your particular work environment, get another product. You're in for big problems if you don't.

Document Control

Finally, a word about document control. Documents have a life of their own. They grow, reproduce, and roam around the system with very little discipline unless you and your network administrator exercise control. As a starting point, you need to have two things for efficient document control. The first is a naming convention and the second is a defined location for storing documents.

Figure 2.
A Database Server Configuration



The naming convention allows easy searches and instant recognition of documents if they stray from their prescribed locations. The defined storage location makes it simple to contain their wanderings and easily locate documents when you need them. Finally, you need a method for periodically archiving older documents and removing the backup files created automatically by most programs when they're no longer needed. There are two ways to store documents: by topic or all together in a single subdirectory.

When you store by topic (all letters in one subdirectory, reports in another, and so on), the best organization is to create a directory called *documents* or something similar. Under that directory create subdirectories organized by topic or document type. Don't get carried away, though. Each subdirectory you create uses up a fair amount of overhead, even empty.

Spreadsheets on a LAN: A Subset of Databases

There are very few differences between spreadsheets and databases in structure. Both are, basically, tables of data. Both have the ability to be programmed (databases with a programming language like dBASE, spreadsheets with macros). But there is a difference in the way the two are used. Databases are used in a query or data input mode on a record-by-record basis. With databases, you're concerned with the contents of an individual record. Spreadsheets usually deal with an entire file, or worksheet, of data. With spreadsheets, the contents of a cell (which corresponds to a single database field in a single record) is usually meaningless by itself.

However, there are several similarities between the LAN organization of a spreadsheet and a file server-type database. First, your spreadsheet program should reside in a special directory in the applications area. In a subdirectory under the application you should place any spreadsheet templates you want to have available for your users. In order to ensure that users don't accidentally alter standard templates, you make that subdirectory read only. This means that no user can write data to the templates. However, they can copy the blank template to another directory for filling and using.

Also in a subdirectory, under the program directory, you should (as we did with the DBMS) place public spreadsheet files. If you have similar security considerations for some spreadsheet files as you do with databases, you can put those in private directories or, if only one person needs access, in that user's home directory.

Printing Problems

There are a few considerations that are consistent with the other types of applications we've discussed. Although not mentioned earlier, there's a potential for difficulties when attempting to print spreadsheets (or, indeed, any application document) on a LAN. You should ensure that your network printer is properly configured and that the application has been configured for the correct port and the proper printer type. Remember, most applications, especially spreadsheets, cannot be set up for multiple printer types. They can only handle one type at a time, so you'll want to pick the optimum printer for the task at hand knowing that if you're wrong, you'll have to go to the trouble of configuring your spreadsheet program again.

Finally, you'll want to consider that most spreadsheet programs keep track of the users under a particular multiuser license. That means that, like word processors, there might be a requirement for individual user configuration (profile) files. If that's the case, put them in the users' home directories.

E-Mail, Menu Shells, and Other Miscellany

Although many networks come equipped with e-mail and menus, some do not or else the programs supplied aren't adequate for your needs. In that case you'll certainly want to explore third-party e-mail and menus. Some are very good; others are expensive packages that really don't add much to the network. We'll discuss internal utilities such as these in the next section. But first, let's look at the third-party offerings to try to determine what makes a good third-party LAN utility.

Menu Shells

Some menu shells are advertised as groupware but are really little more than menus and a few simple utilities. There are a few of these, however, that are very good and merit consideration.

One such program is *WordPerfect Office* from WordPerfect Corp. This is an extraordinarily simple program. It appears to do very little except provide a menu, some e-mail, and a bit of calendaring. However, that's really where its beauty lies and why it's one of the best examples of a menu shell for LANs around.

There's also an implication embodied in *Office* that you should pick a menu shell with its own e-mail and other utilities included. Actually, this is a fair analysis. If you stack applications, shells, utilities, and menus on top of each other indiscriminately, you'll reap the reward of confusion. Keep it as simple as possible. You'll be surprised how beneficial simplicity can be in an otherwise complex environment.

It's a good idea to provide a menu shell of some type for all users. This keeps users from getting tangled up in DOS directories and the complexity of network drives. But, in order for the shell to be useful, it must have three characteristics. First, it must be very easy to integrate into the LAN and its applications. Second, it must be easy enough to use so that the users won't shun it. Finally, it must not allow escape to the DOS prompt except at the network administrator's discretion.

E-Mail

Regarding the issues of e-mail and network utilities, here are some simple guidelines. E-mail should fit well into the organization and into the LAN. If there's also a calendar as part of the shell or e-mail package, be sure that you understand the ramifications of allowing a computer to schedule time for you and your coworkers. Many users avoid calendaring as much as possible since it tends to be pretty inflexible. As to network utilities, if you're an administrator, be very careful what you allow on the LAN. If you're a user, check with your administrator.

Utilities

Most network utilities have the ability to cause a great deal of trouble on the LAN and should be avoided except by the most experienced of users. In no event, however, should utilities of unknown origin (such as bulletin boards or trading software) be allowed on the LAN. In general, the rule is if it didn't come with the LAN, be sure you need it and be sure it won't damage the network if used incorrectly.

Network Installed Services?

Given the choice, you'll gain one big advantage by sticking with the menus, e-mail, and utilities provided by the network. That advantage is that these services were designed to work on their LAN. You can be certain of compatibility, if nothing else. That's the good news. The bad news is that in some cases the network services leave much to be desired in terms of functionality. There's one area, however, where you'll find that the network vendor probably supplies the best solution. That area is supervisor's utilities. There's a trend, as the new breed of powerful LAN operating systems takes control of the high-end network market, towards very complete and powerful tools for the network administrator.

Supervisor's Utilities

What should you look for in supervisor's utilities? Start with tools that allow easy administration of users and applications. You should be able to add, delete, and modify user privileges easily and quickly. The tools should be menu driven without sacrificing completeness for simplicity. You should also be able to install new applications and modify existing ones with little effort. If at all possible, you should be able to perform these routine tasks from locations other than the network console and without bringing down the network for more than a few moments at most.

The next tools you'll want are those that allow you to monitor network activity and tune the LAN as loads increase. You'll want to know when response time on the system is becoming unacceptable, and you'll want the tools to determine why. Finally, you'll need tools to help preserve the security of the LAN. These include audit trails for both users and programs, the ability to observe activity on the network, and logs of both successful and unsuccessful logon attempts. You'll want to be able to change passwords and restrict use of selected files by selected users at selected times, or any combination thereof. In short, your security toolkit should let you know precisely who is doing what on the LAN. Once you know, it should help you control the activity.

Third-Party E-Mail

Since there are new e-mail standards evolving, you need to decide what level of compatibility you need with the outside world. In many cases, because compatibility is such a serious issue, the LAN developers leave the e-mail decision to the user. One such network is Novell (in version 2.12 and higher). Other networks, like 3Com, stick to the current standards and provide e-mail.

When you're deciding what e-mail system you want for your network, the first question to ask is with whom are you going to communicate. If the answer is that all your communication will take place within your single networked workgroup, the LAN e-mail system is probably just fine for most applications. If, on the other hand, you'll need to communicate with other external systems, take a hard look at what those systems require and act accordingly. E-mail compatibility questions will be the determining factor surrounding your choice of network vendor-supplied mail packages or third-party products.

Summary

In this report we've discussed the issues that surround the use of applications of various types on a local area network. We've specifically discussed some representative categories of applications in the context of how they fit into a network scheme. While we couldn't show specific examples due to the differences between various network operating systems, we did discuss the requirements for databases, word processors, spreadsheets, and peripheral applications such as e-mail.

There are several other types of applications you're likely to see on networks. However, you'll find that most of them are simply permutations of one of the "basic three" classes of user programs. The key, as we have shown, is to keep application and data separated; keep order in the way you locate applications and data; where possible, adopt file naming conventions that allow you to quickly and easily locate files; and be careful with pathing, to ensure that applications can find their ways to their associated data and user profiles.

A final word about networked applications. Some applications allow you to connect more users to the application than your license allows. There are two reasons not to do this. The first reason is that you're violating your license agreement, which amounts to software piracy. The second and probably the more persuasive reason is that occasionally an application simply won't work properly if it isn't used properly. You might be able to "get around" the user limits, but those limits are often imposed for more than just marketing reasons. Don't take chances.

Use properly licensed multiuser applications on your LAN and install them properly, within the limits of your user license and with the user registrations filed. User registrations for most multiuser applications are filed by the network administrator. Filing the user registration usually assures you of good technical support. As the supervisor, that means less of your valuable time spent untangling strange problems with an application. As the network user, it means good support all up and down the line, from you to your LAN supervisor. ■

LAN Internetworking

In this report:

Datapro Summary

No LAN is an island! The standalone PC is almost an anachronism, and isolated workgroups will soon follow. The new requirement is for global information sharing, in which communication must be rapid, contentionless, and error free. Against this model, this report explores state-of-the-art LAN interconnection technologies: LAN expansion, bridging, and routing; and other techniques for interconnecting remote LANs over metropolitan and wide areas. Lower level devices, such as repeaters and bridges, are less risky, cheaper, and support higher network performance. Network-layer routers are costlier and exact performance penalties, but provide much greater operational functionality. Each interconnection scenario has its own set of special requirements that determine the suitablility of any given solution.

Were there any visionaries in the early 1900s who predicted the role of information in today's industries? As late as the 1960s, would anyone in their right mind have suggested that the power of a first-generation computer would be increased by many orders of magnitude and packaged in a device one could hold in the palm? It wasn't that many years ago when passing magnetic tape was the primary interface between batch processing systems in a corporate data center. In the early days of LANs, there were only a few small voices urging the industry to think of LANs—not as isolated collections of computer resources, but rather subnets within the global internet (see Figure 1).

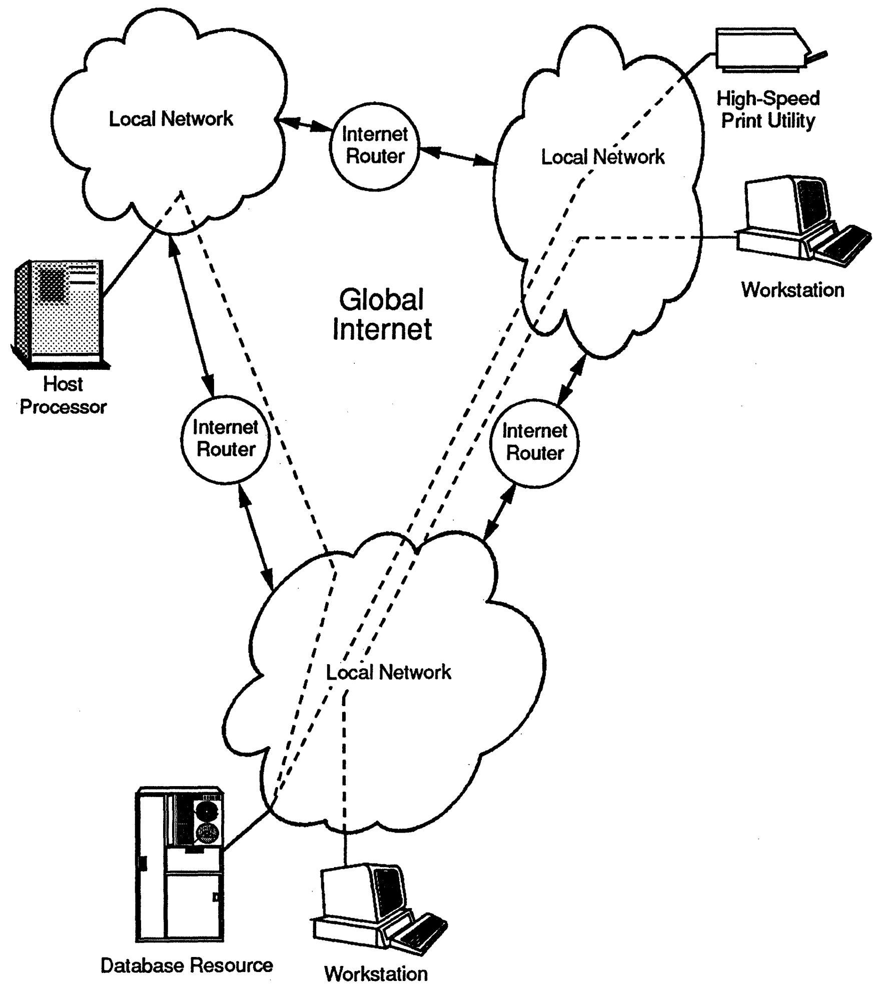
—By Michael L. Rothberg
President
Applied Network Solutions, Inc.

The Need for LAN Interconnection

Perspectives are changing. The standalone PC is almost an anachronism, and the isolated workgroup LAN is soon to follow. No LAN is an island! For network architects to develop comprehensive solutions to organizations' computer networking requirements, they must treat the requirements as "global."

While it is reasonable to expect a workstation user to share files with other users in the same room, it is also reasonable to expect global information sharing. For example, a bank customer at an Automatic Teller Machine (ATM) in New York should be able to access an account database on a LAN in Omaha, NE,

Figure 1.
The LAN in the Internet



while another ATM customer in San Francisco simultaneously accesses the same database. The important requirement in this scenario is that neither user should perceive the database as anything other than a local, privately owned resource. For this to become a reality, communication must be rapid and error-free with no contention for any shared resources.

This report's objective is to explore the state-of-the-art of various LAN interconnection scenarios and to measure them against this model. These will range from physically expanding a single LAN, to "bridging" multiple, collocated, independent LANs, to interconnecting geographically remote LANs.

One should bear in mind that as users, we are "pushing" the state-of-the-art in terms of internetworking, and the most desirable transparent solutions may not always be available. This report highlights real solutions and indicates emerging and future solutions accordingly.

Approaches to LAN Interconnection

One of the greatest obstacles to understanding technology is the lack of standards for jargon. The world of internetworking is no exception. While the terms and definitions we are about to stipulate may raise your eyebrows because of past usage patterns, this is in fact the direction in which the standard definitions are moving (see Figure 2):

1. Physical Layer Relays—Repeaters or amplifiers that simply extend the distance a signal can be driven within a single network. These devices rarely play any role in protocol operation or conversion, although in the case of IEEE 802.3 "Ethernet" there are smart repeaters that insulate consecutive cable segments from faults on other segments. (For our purposes in this report, we are using the terms Ethernet and IEEE 802.3 interchangeably. While early versions of the proprietary Ethernet product

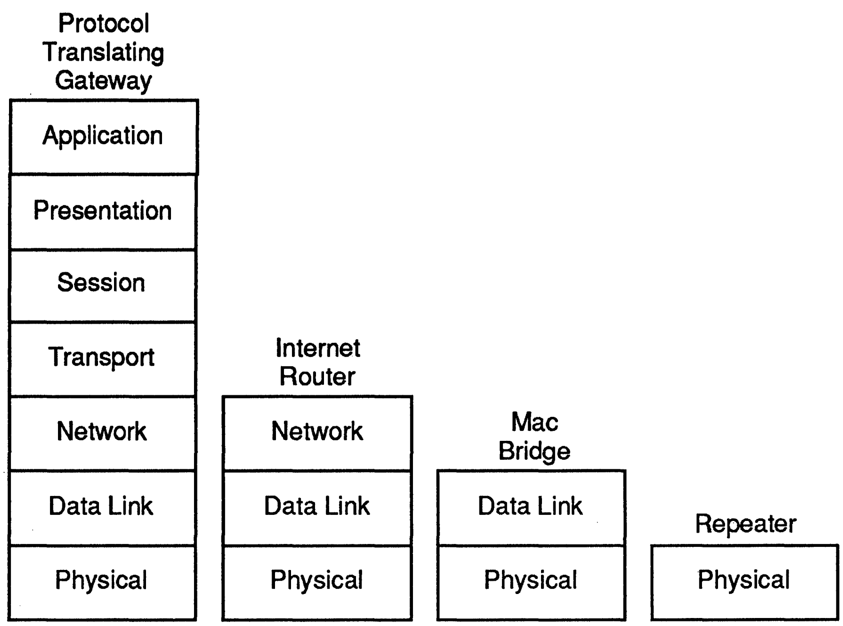


Figure 2.
Network Interconnection Levels

were not compatible with IEEE 802.3, it is often easier to recall an Ethernet model as opposed to translating terminology such as 802.3.)

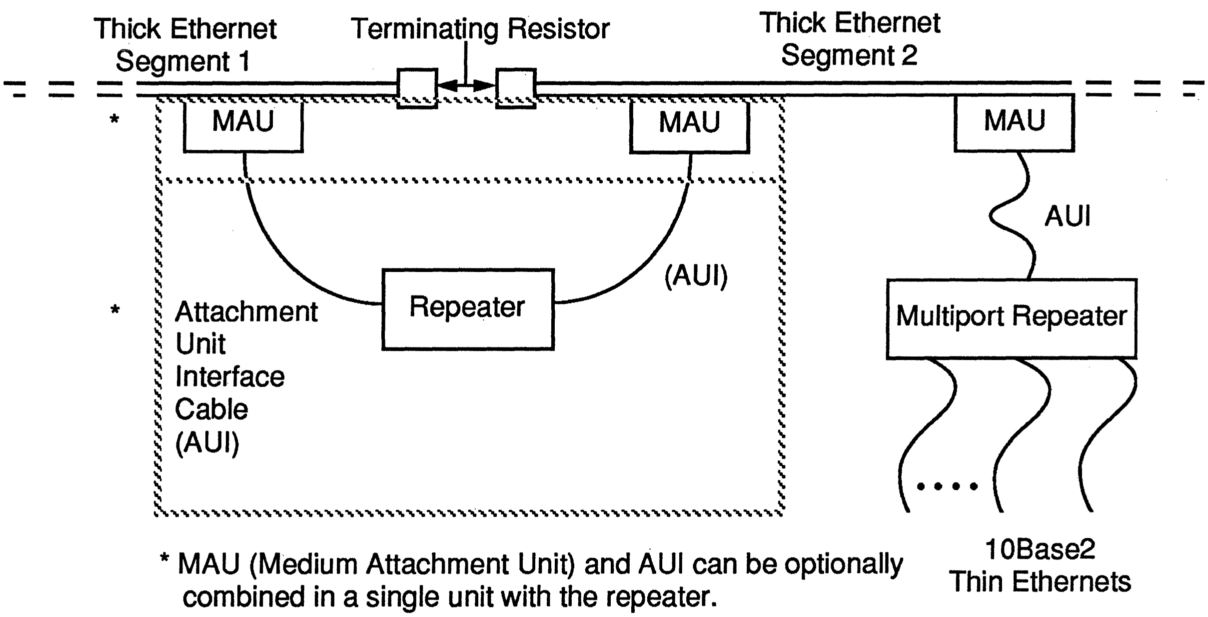
2. Data Link Layer Relays—Medium Access Control (MAC) Bridges and Bridge Routers (Brouters). These devices perform minimal protocol conversion between similar networks, where all higher layers are common implementations.
3. Network Layer Relays—Internet Routers that convert all protocols at the lower three layers. There may be significant delay factors depending on the implementation and selected options. A number of devices are marketed combining the functions of Data Link Layer bridges and Network Layer routers. They are known as brouters, but are not discussed as a separate category.
4. Metropolitan area networks (MANs) and Fiber Distributed Data Interface (FDDI)—These networks are often used to provide connectivity between LANs spaced at moderate distances from one another, as opposed to bridges connecting co-located LANs and routers connecting distant LANs.
5. Application Layer Relays—Protocol Translation Gateways that actually map application functions as well as all of the supporting lower layer implementations. These devices are beyond the scope of this report; however, the features providing inter-LAN connectivity are dealt with in the discussion of lower layer relays.

In keeping with our topic, interconnecting LANs, this report focuses on those alternatives most relevant to the problem. Bridges and repeaters are the most commonly implemented solutions in the LAN world. Internet routers are less often used to interconnect local networks directly, but become an important option when interconnecting remote LANs through a public packet switched network. Protocol translation gateways at the Application Layer are rarely used to interconnect LANs, but are important in the context of providing interoperable solutions between organizations.

In evaluating each of these four alternatives for LAN interconnection, it is imperative to consider several important issues and features:

- Connectivity—Physical interconnection options and topological considerations.
- Transparency—The extent to which protocol overhead is imposed in mapping dissimilar features and functions of the connected networks.
- Imposition of delay—The effect of throughput degradation on different types of traffic scenarios.

Figure 3.
Repeaters



Repeaters in the 10BASE5 and 10BASE2 environments.

- Ease of implementation—Capability for devices to provide a level of interoperability, facilitating multivendor implementations.
- Standards availability—The status of standards for various internetworking facilities.
- Cost—Developing and procuring hardware and software interconnection resources.

Repeaters

The most popular and well-known LAN repeaters are those used for baseband Ethernet and IEEE 802.3 implementations (see Figure 3). The most common implementations of these repeaters are those employed in the traditional "thick" Ethernet or IEEE 10BASE5 systems. Other multiport repeaters are available for connecting "thin" Ethernet or IEEE 10BASE2 segments to a larger 10BASE5 backbone.

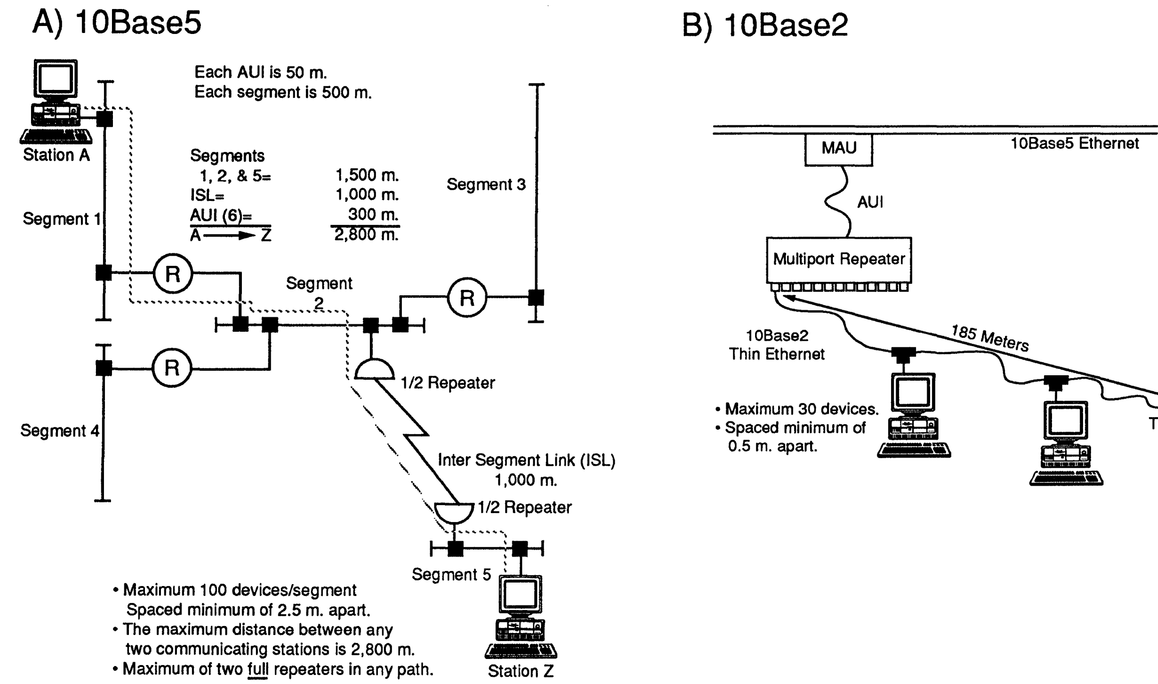
We will not address repeaters in 10BASE"T" twisted-pair Ethernets, since these systems are constructed on a hierarchical hub principle where each hub is an active repeater/concentrator. Likewise, we will not address repeaters in the Token Passing ring, since each station in the ring provides a repeater function. Thus, using repeaters in these environments is strictly for rebroadcasting signals to the next station, as opposed to physical expansion as in a bus network.

Repeater Functions

The repeater's primary purpose in baseband implementations is to regenerate the signal from one cable segment to the next. This effectively enables one to expand the network, thus increasing the overall distance between communicating stations and also increasing the number of network terminations. Theoretically, one could extend the network infinitely by adding repeaters and additional segments.

There are minor delays associated with the repeaters, however, as well as propagation delays on the cable itself. These delays adversely affect the access protocol (carrier sense multiple access with collision detection). To remedy this, design constraints are imposed upon this type of LAN relating to the number of repeaters and the number of cable segments (see Figure 4).

Figure 4.
Ethernet Constraints



Ethernet/IEEE 802.3 (10BASE5/2) design constraints.

Ethernet Design Constraints

As illustrated in Figure 4, the maximum distance between any two communicating stations on a 10BASE5 Ethernet is 2,800 meters with no more than two full repeaters in the path. Note that connecting the intersegment link is accomplished by using half repeaters. The total number of repeaters in the path between stations A and Z remains two. Note that the only difference between the intersegment link and the thick Ethernet segment is that there are no devices terminated on the link. This reduces the attenuation by controlling the phenomenon known as *insertion loss.* These limitations are a compromise between general connectivity requirements and network performance requirements.

Using repeaters provides a great deal of flexibility in wiring even the largest buildings, but one must remember that repeaters connect segments and links of the same network. This means that with the exception of some unusual situations, all signals are broadcast on all segments and links through all repeaters.

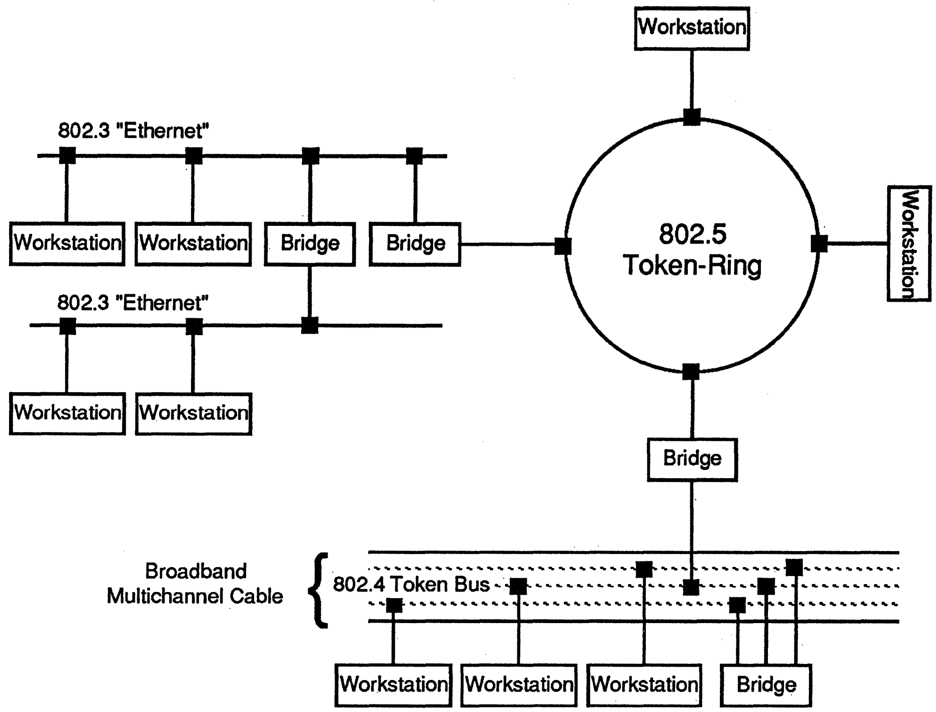
A limitation is the number of workstations that can be physically attached to a single network. A maximum of 100 stations per segment is generally not a severe limiting factor, but when faced with a maximum of 1,024 stations in the entire network, the limitation becomes more of a disability—particularly in large enterprise networks. In the early days of LANs, the solution to this problem was to install a broadband LAN that yielded significantly greater bandwidth, distance, and number of terminations. Until the advent of Medium Access Control (Data Link Layer) bridges, this was the only workable solution, despite all of the inherent problems in broadband design, installation, and maintenance.

Repeater Standards

With the advent of the IEEE 802.3 Supplement specifications, standards for 10BASE5/2 repeaters made their first appearance. Some of the additional capabilities included in this specification dealt with fault isolation issues. The latest repeater designs isolate cable faults to a given segment. For example, the repeater will partition itself from the next segment if it detects too many consecutive collisions on the prior segment. The repeater will also isolate the "offending" segment if a "jabber

Figure 5.
Bridging Incompatible LANs

Bridging compatible and "nearly" compatible LANs.



control" failure is detected in any device. The basic theory applied here is that it is better to lose a segment than the entire network. It is also reasonably likely that nodes most often communicating with each other are probably within the same workgroup and, as such, on the same segment. The result, then, is to maintain communications between some portion of the population, while other remote services become temporarily unavailable.

Implementation and Cost

While repeaters are generally easy to install, users must maintain accurate cable plant management records. Most networks are the result of evolutionary pressures, and frequently depart from the original design. While the network architect planned the topology and configuration with maximum flexibility in mind, these design guidelines were most likely compromised within the first six months of the network's operation. The result is simply that the network may not be configured the way you expect it to be. Adding a repeater might, in fact, create a configuration violation that detrimentally affects network performance. It is, therefore, essential to keep configuration records, preferably in an automated form.

Repeaters are the least expensive alternative to LAN expansion. Typical midrange repeaters for 10BASE5 LANs cost between $1,200 and $1,500. Certainly, one may find products that fall above or below this range, but not with any significant deviation.

Medium Access Control (MAC) Bridges

Connectivity Issues

MAC bridges or Data Link Layer relays connect separate, independent LANs. The degree of performance transparency in accessing resources across the bridge is primarily a function of contention for the bridge and application resources. Bridges are primarily used to connect compatible and "nearly" compatible LANs (see Figure 5). In this context, we

Table 1. Maximum Frame Sizes for IEEE 802 LANs

LAN Type	Speed	Maximum Frame Size
802.3 (10BASE5)	10M bps	1,518 octets
802.4	10M bps	8,191 octets
802.5	4M bps	4,500 octets
802.5	16M bps	18,000 octets

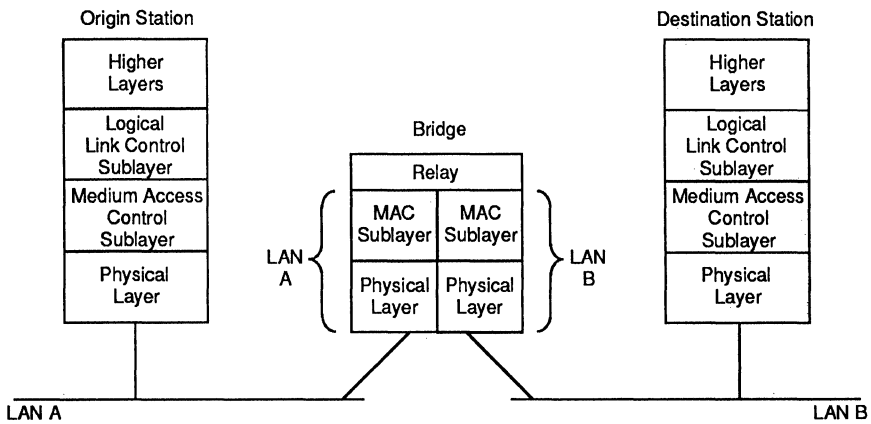


Figure 6.
MAC Bridge

MAC bridge layered architecture.

define compatible LANs as different versions of 802.3 Ethernets (e.g., 10BASE5/10BASE2), while nearly compatible LANs might be 802.3 Ethernets and 802.5 Token Passing Rings.

Layering in Bridges

In the LAN environment, bridge software usually resides at the lower half of the Data Link Layer, the MAC sublayer. The upper half of the Data Link Layer, the Logical Link Control (LLC) sublayer, is common to all standard LAN implementations, and is consequently not involved in the bridging function (see Figure 6).

The more common implementations of bridges today employ topologies known as *spanning trees*. Spanning tree networks are characterized by the fact that there is only one path between any two communicating nodes (see Figure 7). A spanning tree topology permits neither closed loops nor alternate paths, as these would require additional routing intelligence. Spanning tree bridges lend themselves to bus-oriented LANs due to the structured nature of their topologies. Implementing a closed loop in a spanning tree environment will result in an infinite replication and broadcast of packets through the loop, eventually bringing the network to a screeching halt caused by congestion.

Bridge Features and Operation

Addressing Transparency in Learning Bridges

Bridges are generally transparent at the source and destination hosts in that packets are not addressed to the bridge. They effectively operate as store-and-forward switches between LANs, but to provide a minimum degree of operational efficiency they must be imbued with some intelligence. Source routing bridges are exceptions to this rule, in that the packet contains a special routing information field that identifies the bridge and network segments. We discuss source routing bridges in detail shortly.

Congestion Avoidance/Filtering

Bridges not only provide connectivity between LANs; they also avoid serious congestion problems between LANs by filtering traffic and forwarding

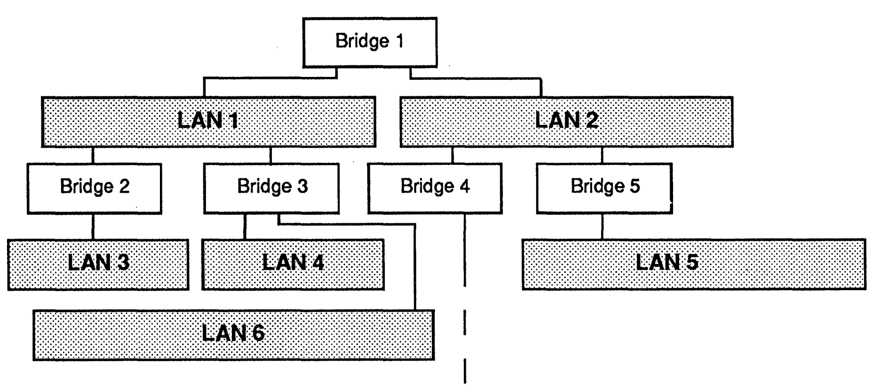


Figure 7.
Spanning Tree

Bridged spanning tree LAN.

only that addressed to the destination LAN. Imagine for a moment that two interconnected Ethernets each currently support 6M bps in real traffic. If the bridge was to forward all traffic, each Ethernet would be required to support 12M bps—clearly an impossible task for a 10M bps LAN. If only 10% of the traffic from each LAN was destined for the other, however, the result employing filtering would be a 6.6M bps load on each.

To accomplish this task of filtering, bridges must have the capability of "learning" where nodes are located based upon their addresses. They can then forward only those packets that are necessarily transmitted to the destination network.

The learning process is a clever phenomenon. When a bridge receives a packet, it searches for the origin (source) address in a routing table. If the address is not in the routing table, the bridge adds it. Thus, if the bridge ever receives a packet from its other side destined for this source address, it can now determine whether to forward the packet by examining its routing table. So in essence, the bridge learns from source addresses as they go by, and it applies this knowledge to forwarding decisions (filtering) on destination addresses.

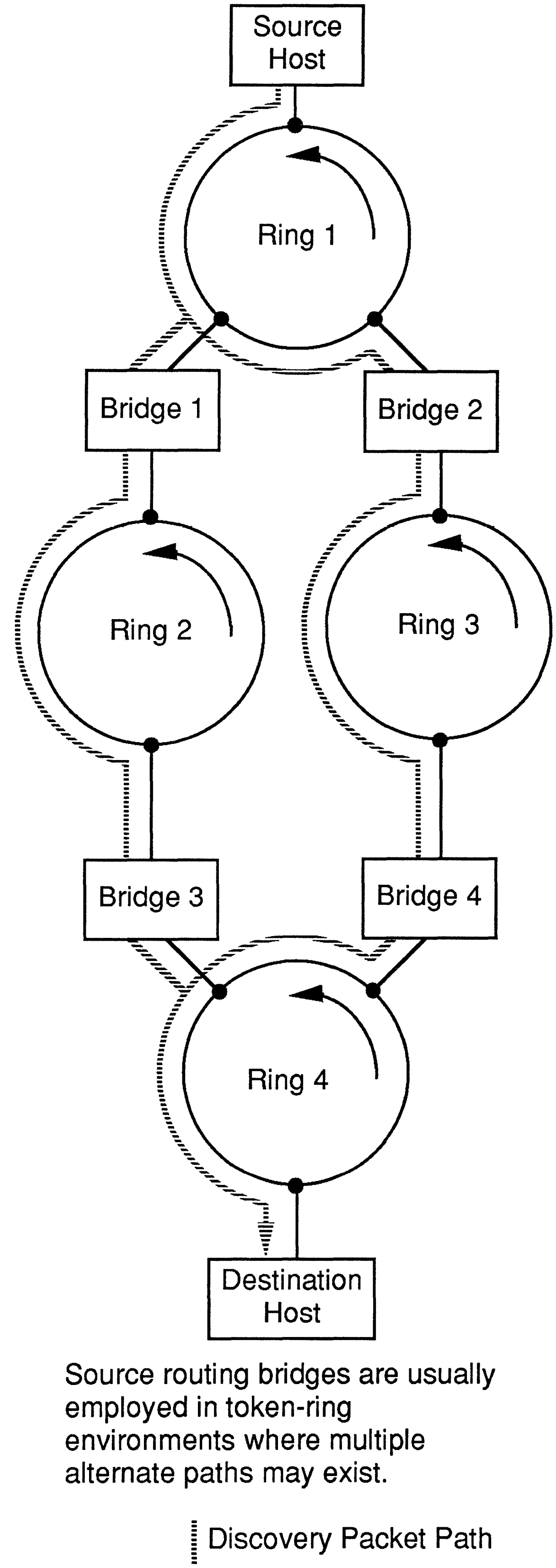
The question then arises: What does the bridge do with a destination address that it has not yet learned? In this case, the bridge will broadcast the packet on all of its ports (except the one through which it was received). As the bridge learns, the need for broadcasting will diminish, except for those broadcasts that are mandated by a packet's broadcast destination address or those resulting from inserting new stations on any of the connected LANs.

Secure Compartmentalization

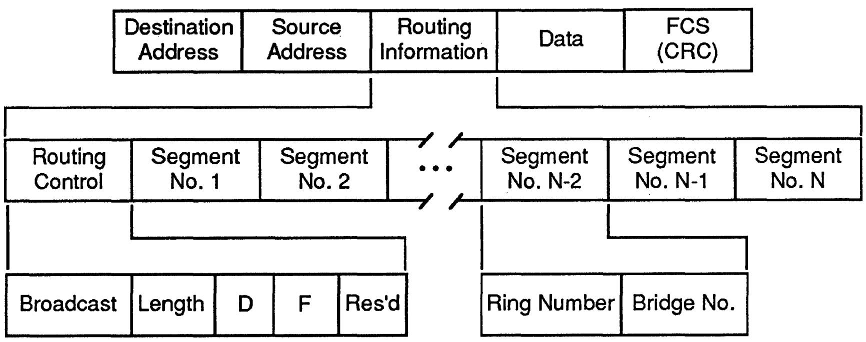
Bridges can provide other operational features. In a secure or sensitive environment, they can evaluate source and destination addresses to ascertain whether the communication is permissible. Secure "compartmentalization" of network resources is a basic requirement in these environments.

Based upon input from the LAN administrator, the bridge can filter packets, preventing unauthorized packets from reaching the destination LAN. The compartmentalization can encompass an entire LAN, or simply a specific server resource on that LAN.

Figure 8a.
Source Routing



Source routing ring configuration.



Length = Length of the routing field including this control info.
D = Direction from originator or return.
LF = Largest frame supported by the bridge.
Res'd = Reserved.

Figure 8b.
Source Routing

Source routing header information.

Other MAC Bridge Issues

Bridges are not panaceas. They do not perform magic, and as such present some limitations in LAN interconnection.

Addressing

For bridges to recognize address and filter packets, all addresses must be the same length. While this is not an issue in IEEE 802 LANs, other proprietary approaches may present some incompatibilities that must be resolved. Earlier versions of the IEEE 802 specifications permitted both 16-and 48-bit addresses. While 48-bit addresses are predominant, there may be some residual 16-bit implementations (although unlikely).

In addition, all addresses between the interconnected LANs must be unique.

Priorities

Priorities present some special concerns. For example, IEEE 802.3 LANs have no priority mechanisms. 802.5 token-rings employ a fairly robust scheme enabling high-priority applications (such as token management) to preempt the next rightful user. 802.4 token busses employ a contention resolution scheme also ensuring that all users are provided with fair access. Imagine a scenario where two 802.5 token-rings are remotely connected via an 802.3 Ethernet. The priority information from the source token-ring LAN will be lost upon arrival at the destination LAN.

This requires that packets from the source ring be assigned a default priority. Since "ring management" and "token management" are generally limited to the specific LAN, this issue may be academic. If an application resource has a high priority across both the origin and destination LAN, however, that priority information will be unrecoverable.

Broadcasting

Depending on configuration parameters, bridges may or may not be capable of broadcasting packets. In many cases, however, bridges can be configured to broadcast all packets based upon the broadcast address, or selective packets where the packet type and the broadcast address are considered. As determined previously, broadcasting can have a detrimental effect on the traffic loads of the interconnected LANs. Repeated attempts at broadcasting may result in a phenomenon known as a "broadcast storm" or a "LAN meltdown." As the traffic increases beyond the LAN's capacity, all meaningful communication is eventually shut out.

Routing Issues in Bridged LANs

As mentioned previously, bridged LANs employ two forms of routing:

- Spanning tree routing
- Source routing

Spanning tree routing requires that the bridges learn the proper routes and, as such, the source hosts are freed from any routing responsibility. Source routing bridges, on the other hand, need know nothing about the routes, but the source host itself must be significantly more intelligent. The source host broadcasts a "discovery" frame through all bridges on its "home" LAN. As the discovery frame travels through the LANs, it is broadcast through all other bridges. Each bridge makes a notation in the header of its ring number and bridge number (see Figure 8). When the packet reaches the destination host, it is returned to the source host via the path in the header. The source host then selects the best path for subsequent transmissions to that destination. Determining the best path may be influenced by transit time as well as the number of bridge hops in a path.

As a result of this process, the source host also learns the maximum acceptable packet size of the destination and transit LANs. This information is stored in the discovery packet's routing control field. This issue becomes critically important when communicating between token-rings and Ethernets.

The maximum frame size parameters vary with different implementations of these networks and often result in requirements for segmentation and reassembly. Although maximum frame sizes are not specified in the IEEE 802.5 token-ring standard, and are specified in other 802 standards, the following de facto guidelines are commonly followed:

While source routing is the approach favored by the 802.5 working group, spanning tree routing is the favored approach in the 802.3 world. The IEEE 802.1 committee, which is responsible for the recently published standard (802.1D), has adopted the spanning tree approach. Token-ring LANs can employ source routing between themselves, but when communicating with other LANs (802.3/802.4), they must employ spanning tree routing.

The trade-off, of course, is that while source routing bridges can be relatively simple, and will impose minimal frame processing, hosts in this environment must process every discovery frame and return them to the sender.

Implementation and Costs

Many of the same LAN topology issues prevalent in repeater placement must also be considered in "bridging." While bridge placement in a source routing environment is more flexible, since loops or alternative paths are expected and can be accommodated by the protocol, in spanning trees "looping" through alternate paths can have a devastating effect, as previously demonstrated.

Bridges are also significantly more expensive than their repeater counterparts. They may start at prices as low as $1,500 for local Ethernet-to-Ethernet connection, but can rapidly climb to $22,000 for LAN-to-Fiber Distributed Data Interface (FDDI) connections. It is not unreasonable to expect prices in the range of $7,500 to $10,000.

Network Layer Relays—Routers

Routers are freestanding computers providing the capability to interface very different types of networks, as well as the capability to determine the best "path" or "route" to get to the destination. Most routers in use today employ the ARPANET Internet Protocol for network interconnection (usually associated with TCP/IP—Transmission Control Protocol/Internet Protocol). They were originally conceived and are useful for interconnecting wide area networks (WANs), where the path through the origin, transit, or destination networks was of little concern to the router. The router was only concerned with the "internet" addresses, which were created at the upper portion of the Network Layer—at the internet sublayer (see Figure 2). The result was that an internet path specified the origin host, the intermediate routers, and the destination host. The individual network topologies were invisible to the router.

OSI vs TCP/IP Routers

With the advent of OSI, more robust internet protocols have become available, but TCP/IP still maintains a dominant market share. The industry consensus is that, ultimately, TCP/IP will be replaced by the equivalent OSI protocols, but most participants are hesitant to predict when. Fortune-tellers like to be paid in advance, which tells us something about the certainty of their art.

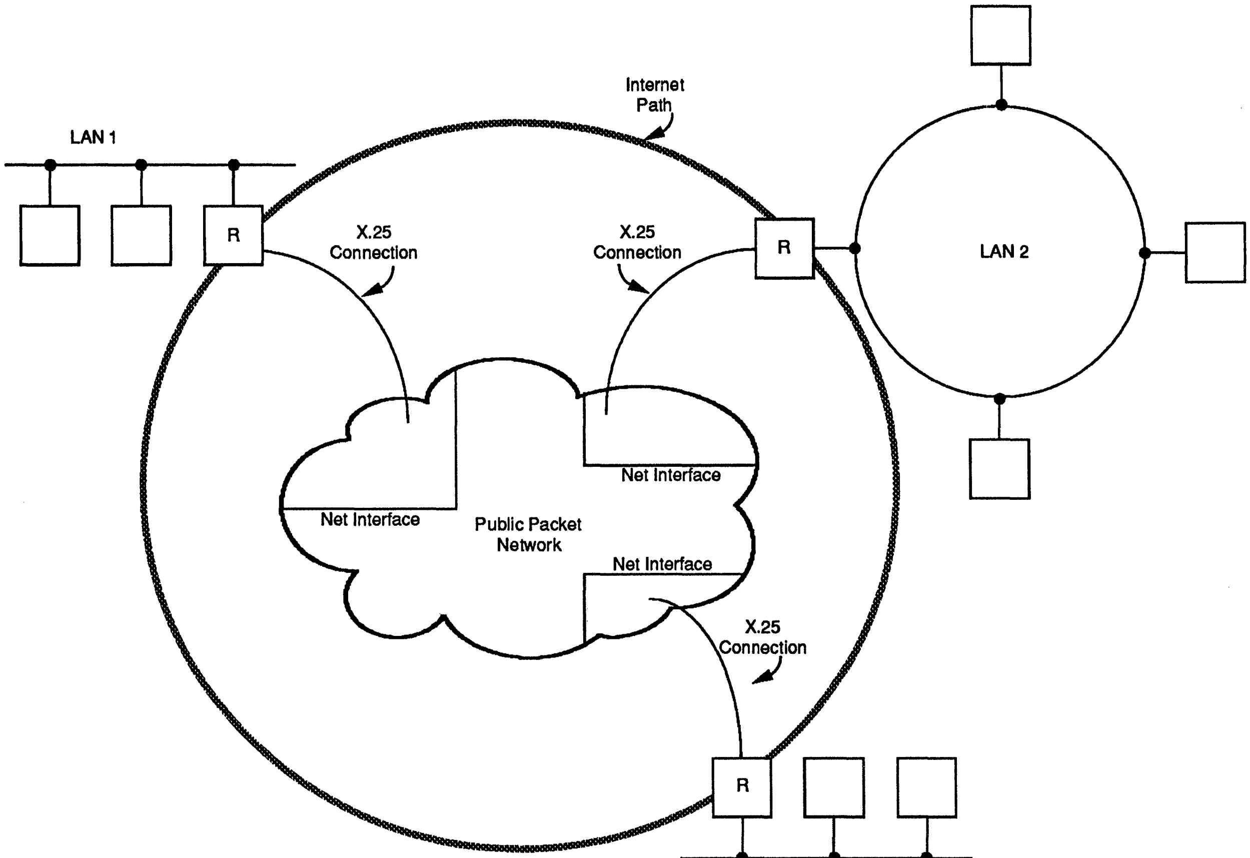
X.25/X.75 Routers

In addition to the OSI and TCP internet protocols, public networks that provide X.25 access also offer a connection-oriented internet protocol known as X.75. While this is most often used in an international environment between public data networks (PDNs) in different countries, it does provide a common approach for PDNs in a domestic environment.

Adapting the Internet for High-Performance LANs

More recently, routers have played an important role in interconnecting remote LANs via a WAN(s) (see Figure 9). One of the difficult issues associated

Figure 9.
Internetworked LANs



LANs across the Internet.

with using internet protocols is that while LANs usually operate at multiple megabits per second, packet switched WANs are generally limited to 64K bps for a given connection. The WAN then becomes a rather significant bottleneck, unless many 64K bps connections are used in parallel.

Frame Relay

Approaches to resolving this problem have yielded new forms of packet-switching protocols such as frame-relay and fast packet-switching service. When comparing these technologies, it is important to remember that the term "fast packet switching" is a generic term encompassing a variety of improved packet technologies, including the following:

- Frame relay
- Broadband ISDN
- Cell relay
- Asynchronous Transfer Mode (ATM).

The term "FastPacket" is a proprietary term and a registered trademark of StrataCom Corp. This technique employs T1 facilities, where the entire bandwidth of the T1 facility (1.544M bps) is allocated to the packet transfer, as opposed to the traditional circuit switched T1 subchannel of 64K bps.

Frame-relay service, on the other hand, is more closely related to conventional packet switching in an X.25 access network. It is a standards-based technique that although not designed to provide LAN interconnection capabilities through the WAN, has been adapted to efficiently provide this service. Current frame-relay versions provide up to 2M bps transfer rates with relatively low overhead compared to X.25. While the skeptic will retort, "2 megabits to connect 10M or 16M bps LANs?", the fact is that the router will filter packets based upon their destination internet address. The result is that only a small percentage, perhaps 20%, of the data will necessarily travel across the WAN. It reduces the overhead of this transmission by gambling on the network's reliability and reducing lower layer error detection and correction. A reasonable gamble indeed, since today's high-speed digital networks are inherently reliable. When an error does occur, the correction overhead is higher since it must be performed at a higher layer (e.g., Transport). This is prudent because unlike, the old axiom, modern packet-switching technologies involve "pounds of prevention vs ounces of cure."

Frame relay was originally designed as an ISDN bearer service for multiplexing individually addressed information units at the Data Link Layer as opposed to allocating a channel through T1-type time slots. In the context of LAN/WAN interconnection, it resides at layer two, corresponding to X.25 Frame Level LAPB (Link Access Protocol Balanced):

Frame relay has been implemented in a form called Frame Relay 1, where a Permanent Virtual Circuit (PVC) similar to X.25 is established. In a version to be implemented in the near future, Frame Relay 2, it is expected that Switched Virtual Circuit Service (SVCS) will be available. This will provide for dynamic bandwidth allocation, and will be less costly in the "bursty" LAN-to-WAN traffic scenario.

Due to the very nature of LAN traffic with its large file transfers and short inquiry response scenarios, flow control will inevitably be a problem at times in the frame-relay environment. The American National Standards Institute has offered a solution in the form of the Explicit Congestion Notification (ECN) flow control protocol.

It appears, then, that frame relay as well as the proprietary StrataCom FastPacket protocols will offer some reasonable solutions to connecting remote LANs through the WAN internet. While it may not be a panacea, it combines many of the better attributes of both packet and circuit switching, but one must remember that there is no best solution in the absence of a specific set of requirements.

Figure 10.
Distributed Queue Dual Bus

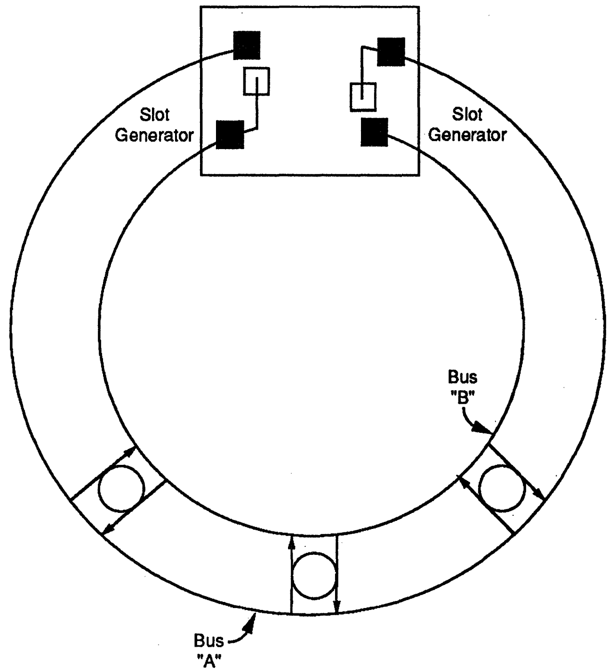


Table 2. Comparing X.25 and Frame-Relay Protocols

Frame Relay	X.25
Higher Levels (TCP/IP, OSI)	Level 3—Packet Level/X.25 Access Protocol
Data Link Layer (CCITT Q.921, ANSI T1.602)	Level 2—Frame Level/LAPB
Physical Layer (X.21—X.21 bis)	Level 1—Physical Level/X.21—X.21 bis

Metropolitan Area Networks and FDDI

The remaining alternative for interconnecting LANs located within a moderate geographic area (on the order of 1 to 200 km.) is establishing a metropolitan area network (MAN) or a Fiber Distributed Data Interface (FDDI). Both of these alternatives are available; however, the FDDI technology is maturing more rapidly simply because of a head start in the standards development process.

MANs

IEEE 802.6 MAN has gone through a number of painful iterations in its development over the past seven years. It originally was envisioned as a CATV broadband network, and from there migrated to a slotted optical fiber ring. More recently, a proposal from Telecom Australia was adopted for an entirely new implementation called a Queued Packet Synchronous Exchange (QPSX). This was then modified and has become known as the Distributed Queue Dual Bus (DQDB). If this sounds confusing, remember that there are two activities that end users should not heed closely. The first is "sausage making" and the second is standards development.

In any case, the basic 802.6 MAN operation consists of two busses designated the "A" bus and the "B" bus. These busses are configured in concentric rings and both are terminated on each end in a scheduling and slot-generation device (see Figure 10).

The nodes form a distributed queue, and making requests on one bus to transmit packets on the other. The architecture lends itself to guaranteed bounds on network access, which, in turn, makes it highly suitable for digital voice traffic. It is an extremely high-speed network, operating at 155M bps.

FDDI

The FDDI optical fiber ring is similar to the 802.5 token-ring in many respects. While it provides a great deal of bandwidth and expanded geographic range, there are limitations that are a function of both the "laws of physics" and specific performance criteria.

The laws of physics require that there be a maximum of 2 km. between nodes because of signal attenuation, with a maximum of three bypassed nodes between active repeaters. Due to clock skew between stations resulting from propagation delay, the FDDI ring is limited to a 4,500-octet packet. Performance constraints dictate that no more than 500 nodes be attached to the ring with a maximum distance of 100 km. overall length.

The frame structure is very similar to the 802.5 ring, and it also supports both 16- and 48-bit addresses. It employs the same 32-bit frame check sequence (Cyclic Redundancy Check—CRC), and a token-passing protocol that appears very similar to the 16M bps token-ring.

The ring is usually wired in dual concentric rings, providing a high degree of reliability as well as facilitating priority access as required.

802 LANs can be connected to the FDDI ring by means of MAC bridges, providing significantly lower performance penalties than internet routers.

Although the current specification calls for 62.5-um. optical fiber, there are implementations (nonstandard) using single-mode, 8-um. fiber. The FDDI protocol has also been implemented on unshielded twisted pair at 100M bps over very short distances.

An FDDI-2 specification has incorporated protocol changes facilitating digital voice traffic, thus making it competitive with the 802.6 MAN.

Summary

There are numerous mature options for interconnecting LANs over both short and long distances. Using lower level devices, such as repeaters and bridges, presents less implementation risk, lower costs, and higher performance. On the other hand, Network Layer routers are considerably more costly, are far from "transparent," but provide much greater operational functionality.

Each interconnection scenario will have its own set of special requirements. It is these requirements that will determine the suitability of any given solution.

One final caution on the subject of standards. While the IEEE has recently introduced the 802.1D MAC sublayer bridge specifications, there are few products currently on the market conforming to these specifications. It is essential that standards compliance be verified before using bridges of different vendor origin in the same environment. There is a strong probability that different manufacturers' bridge products will not interoperate until the standard's acceptance becomes a reality. ■

This report was prepared exclusively for Datapro by Michael L. Rothberg. Mr. Rothberg is president of Applied Network Solutions, Inc., a Somerset, NJ firm specializing in designing, developing, and implementing local and wide area computer networks for government and commercial clients and providing market and product planning services for communications suppliers.

Prior to founding the company in 1981, Mr. Rothberg was a vice president of the Chase Manhattan Bank, NA, where he pioneered the application of digitized speech and local networking technology to support banking applications. He is a frequent contributor to trade publications and is a Datapro advisor.

Planning and Implementing a Local Area Network Internet

In this report:

Datapro Summary

The computer revolution, the personal computer revolution, and finally the local area network (LAN) revolution have changed the way businesses think about the tools of competitive advantage. The next challenge is converting isolated LAN islands into internetworks (internets). Key issues in addressing this challenge include the following: planning a LAN internet; choosing device types; analyzing user needs and traffic patterns; creating topologies and functions to serve specific device types and users in your internet; managing your internet; and preparing for an effective future internet.

The Need for LAN Internets

Interest in LAN internetworking has increased sharply in the past four years in response to new technologies and new business realities. *Fortune* 1000 companies with individual LANs want to leverage their resources by creating linkages, bringing corporate databases within the reach of more skilled workers and bringing a higher level of communication to the company. In addition, the dramatic upsurge in mergers and acquisitions, undertaken to improve competitive positions in the domestic and world markets, has produced merged corporations with incompatible networking infrastructures and an urgent need for effective internets. Since the LAN market has been characterized by innovation and a relatively large number of vendors, the installed base of LANs being challenged by the need for internetworking reflects a significant amount of technical and functional diversity.

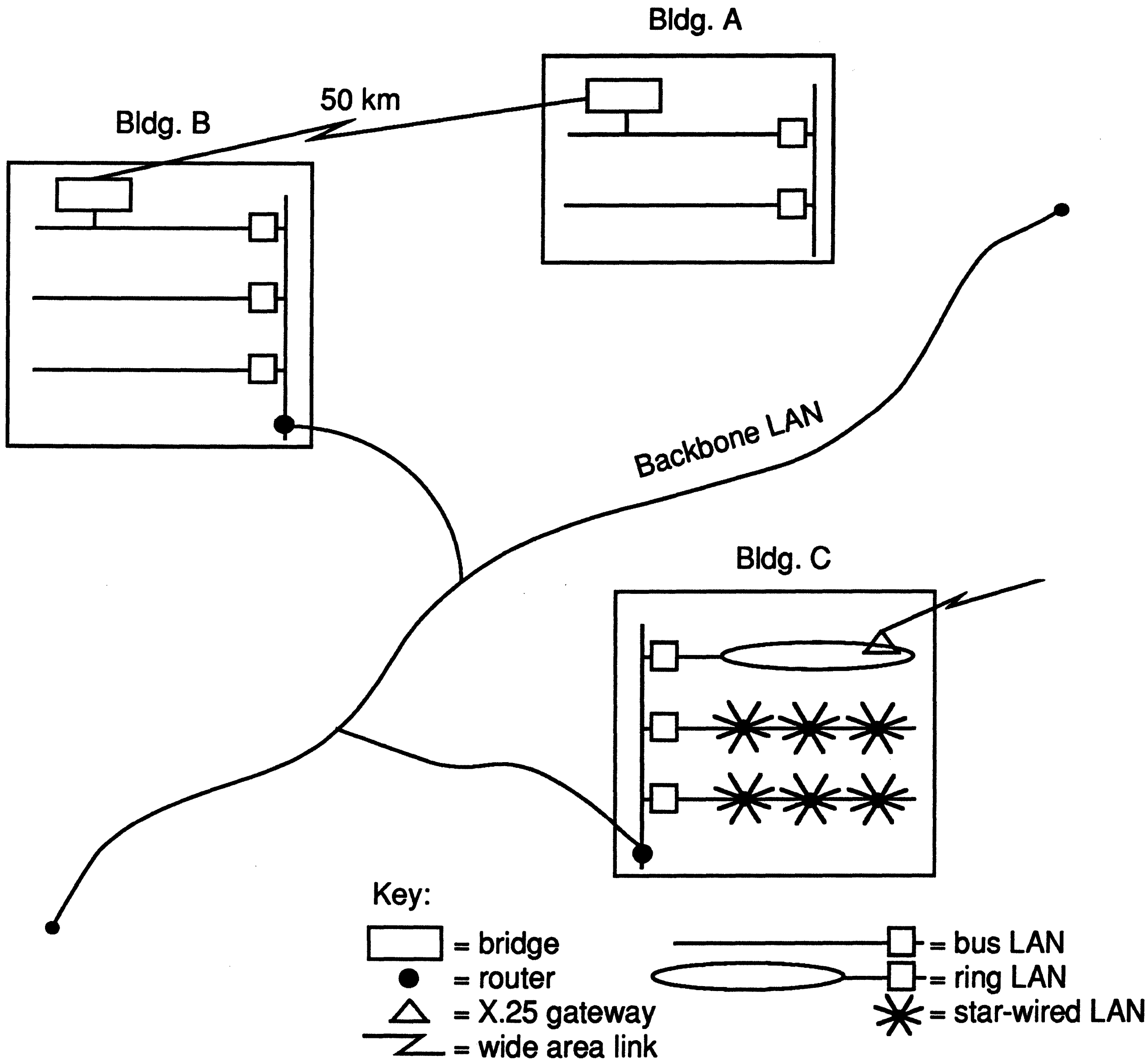
LAN internetworks come in four basic types: local, campus, metropolitan, and wide area. If you link LANs that are within the confines of a single building, you have a local internet. Linking two or more closely spaced buildings creates a campus internet; many users create local internets within each building and then link the buildings together with a high-speed backbone (see Figure 1). Metropolitan area network (MAN) internets give companies a high-speed backbone for connecting LANs across a metropolitan area.

Finally, if any or all of the individual LANs or local internets include wide-area internetworking devices, you can link your local internet to other LANs or internets in other cities, creating a private wide-area network (WAN) with remote internetworking devices. The major difference between a local and a wide-area internet is that a local internet uses a single device to link two or more LANs, while a wide-area internet uses two devices per link because each end of the connection requires its own internet device.

The major desire of LAN users is for transparent access to all resources, regardless of physical location—across the building or across the globe. Incorporating standards into the network design is the most effective way to reach this goal.

—*By Victoria Marney-Petix*

Figure 1.
Campus Internet



A campus internet includes many discrete LANs plus links to the outside world.

Framework for Product Development: Standards Models

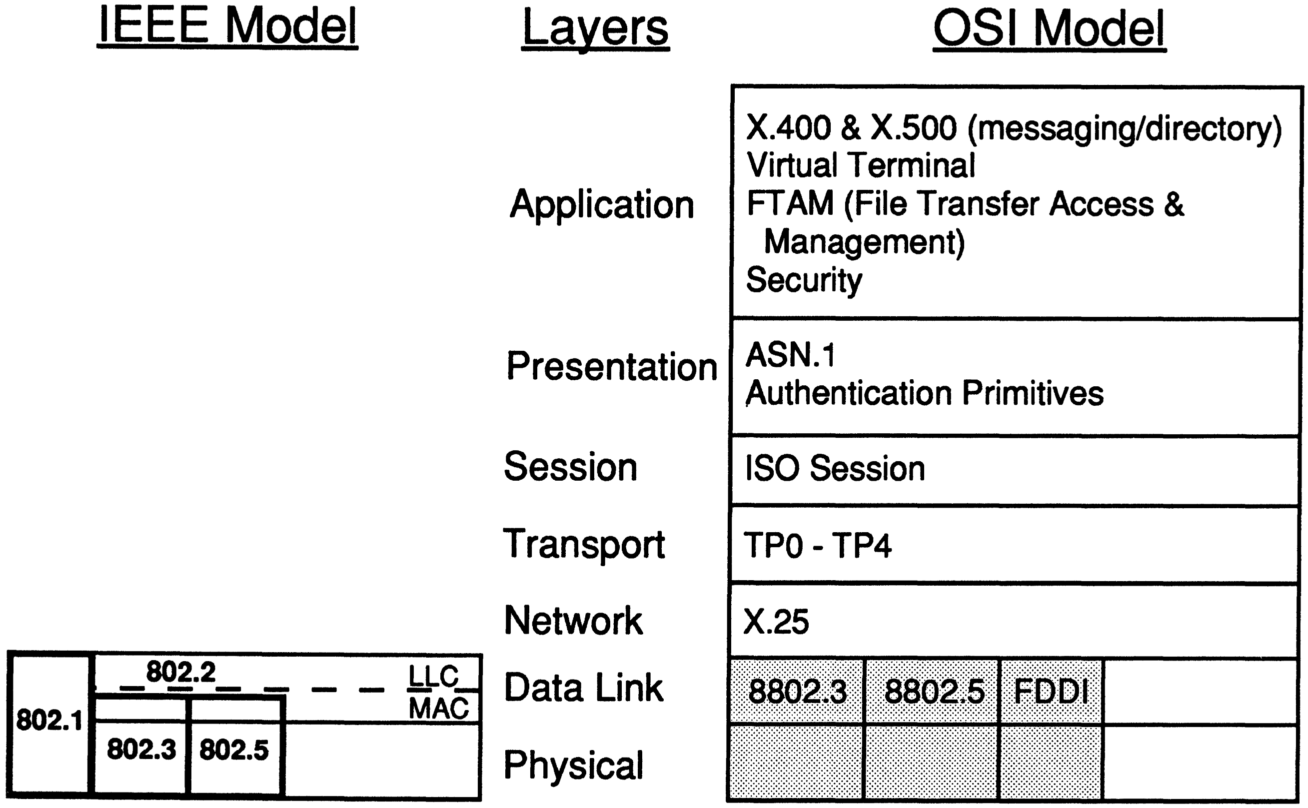
Standards that hold the most interest for internet planners include the IEEE 802 Committee's series of standards and the Open Systems Interconnection (OSI) Model's standards developed by the International Organization for Standardization (ISO) and the U.N.'s International Telegraph and Telephone Consultative Committee for Telephony and Telegraphy (CCITT).

IEEE Standards

The IEEE 802 Committee has developed a series of LAN standards, including the Ethernet look-alike 802.3 standard, the 802.5 token-passing ring standard, the 802.1 standards for network management and internetworking, and the 802.6 MAN standard. The IEEE 802 standards and the first two layers of the OSI model are concerned with the same functions, as shown in Figure 2.

The Physical Layer determines how the data will be placed on the medium, so it specifies voltages, power levels, and physical interfaces between devices and the medium. The medium itself forms a so-called Layer 0 (zero) below the physical layer. The second layer, called the Data Link Layer, has a two-fold purpose in the IEEE version. The Media Access Control (MAC) sublayer helps the first layer regulate how the data will be placed on the medium. The upper sublayer, called Logical Link Control (LLC), creates frames from the raw bitstream of the physical layer. The LLC also determines how a station will get on the network, how it will be addressed, and how the sender and receiver will establish their logical link. It is important to remember that this logical link is a very low-level protocol, analogous to making a physical connection between two

Figure 2.
IEEE and OSI Models



The IEEE 802 standards and the first two layers of the OSI Model address the same functions.

telephones. Logical link control is not concerned with the content of the data that flows over the link.

Figure 2 illustrates that the popular 802.x LAN standards and the ANSI FDDI standard specify the physical layer and the MAC sublayer with the LLC sublayer acting as a single, common specification (802.2) for logical link control.

OSI Standards

The OSI Model divides all the typical network services into seven layers of functionality. Since the Model's purpose is to provide a functional framework for present and future network design and development, each layer is encapsulated with specific functions and specific inputs and outputs to other layers. New protocols can be added to or deleted from the stack without interfering with the functioning of any other layer's protocols.

Above the two lower layers, which are almost exact analogs of the IEEE layers, the OSI model continues with a Network Layer where packets are created and transferred from LAN to LAN. This layer is also called the internetwork layer. Above the Network Layer are the data integrity, transport, data format, security, and authentication functions in the Transport, Session, and Presentation Layers. The topmost layer is called the Application Layer because these protocols provide services to the applications programs themselves, services such as file system management and access (FTAM), virtual terminal, directory (X.500), and electronic mail management (X.400).

Layers and Device Types

There are many different ways to create an internet (see Figure 3). Choosing between the alternatives requires you to understand what kind of internet you want to build and then map those needs to the OSI layers and finally to a specific device type. If you want to link all your LANs into one single enormous LAN, you need Layer 2 linkage, which is called a bridge. If you are primarily interested in traffic management—multiple routes from one LAN to another—and you want to manage each LAN individually, you are looking for a Layer 3 device, a router. If you want your LAN-to-LAN linkages to connect disparate operating systems and application environments, you are looking for a Layer 7 device called an application gateway. If you are looking for a linkage to wide-area packet switching networks, you are looking for a different kind of network layer device, called an X.25 gateway.

One of the most difficult challenges facing an internet manager today is consistent definition of terms; vendors in the internet industry have been particularly flagrant in misusing these terms, either unintentionally or quite intentionally. Network managers must ignore the vendor labels and dig into layer functionality in order to discover whether a particular vendor's "gateway" is really functionally a bridge or a router.

Bridging (Layer 2)

Bridges are very simple to operate, relatively inexpensive, and consequently very popular. Functionally, a bridge

Figure 3.
Layers and Device Types

OSI Layer Number	OSI Layer Name	Device	Unit of Transfer
7	Application	Applications Gateway	Completely formatted packet
4-6	Transport, Session, & Presentation		
3	Network	X.25 Gateway or Router	Packet
2	Data Link	Bridge	Frame
1			

To create the optimum internet, the user must determine the internet's requirements and then map those requirements to the OSI layers and then to a specific device type.

links two or more LANs into a single, larger LAN, performing the linkage at the Data Link layer. The individual LANs are called LAN segments, since the bridges turn the formerly independent islands into a continuous land mass. Since it is a Layer 2 device, the unit of transfer is the frame.

Topology and Spanning Trees

Your topology should never include two or more active bridges connecting the same two segments. In addition, most state-of-the-art bridges conform to the 802.1 Section D Spanning Tree standard, which creates a logical tree structure when it determines the best path from sender to receiver. In order to create a tree from what is usually a mesh, one or more bridges is inactivated, as shown in Figure 4. Spanning Tree was designed to circumvent looping, which can occur in poorly designed LANs where bridges run in parallel; packets can "loop" back to their originating segments, clogging the network and making no progress toward their destinations. In properly designed networks, bridges constantly test for loops and shut themselves down if they discover one.

Managers must understand addressing to interrelate internetworking devices, functions, and business needs, because a network device's address is its identity. Let us look at how bridges read addresses.

Filter and Forward

A bridge reads the data link (DL) source and destination address of every frame that comes its way. As it does, it learns the locations of sender devices. Learning bridges generate these device tables automatically, as shown in Figure 5. If a bridge has not seen a particular address in the source address field for a specific period of time, it deletes it—a process known as aging an address.

When a bridge receives a new frame, it reads the destination address and determines whether the frame is local or foreign, a decision process called filtering. If the destination address is local—on the same segment as the source address—the bridge discards the frame. If the address exists on the second segment, the bridge forwards it; that is, it sends it through. A bridge that can filter at the full bandwidth of the LAN is transparent to end users—they don't know that it's there because they experience no delays in making connections.

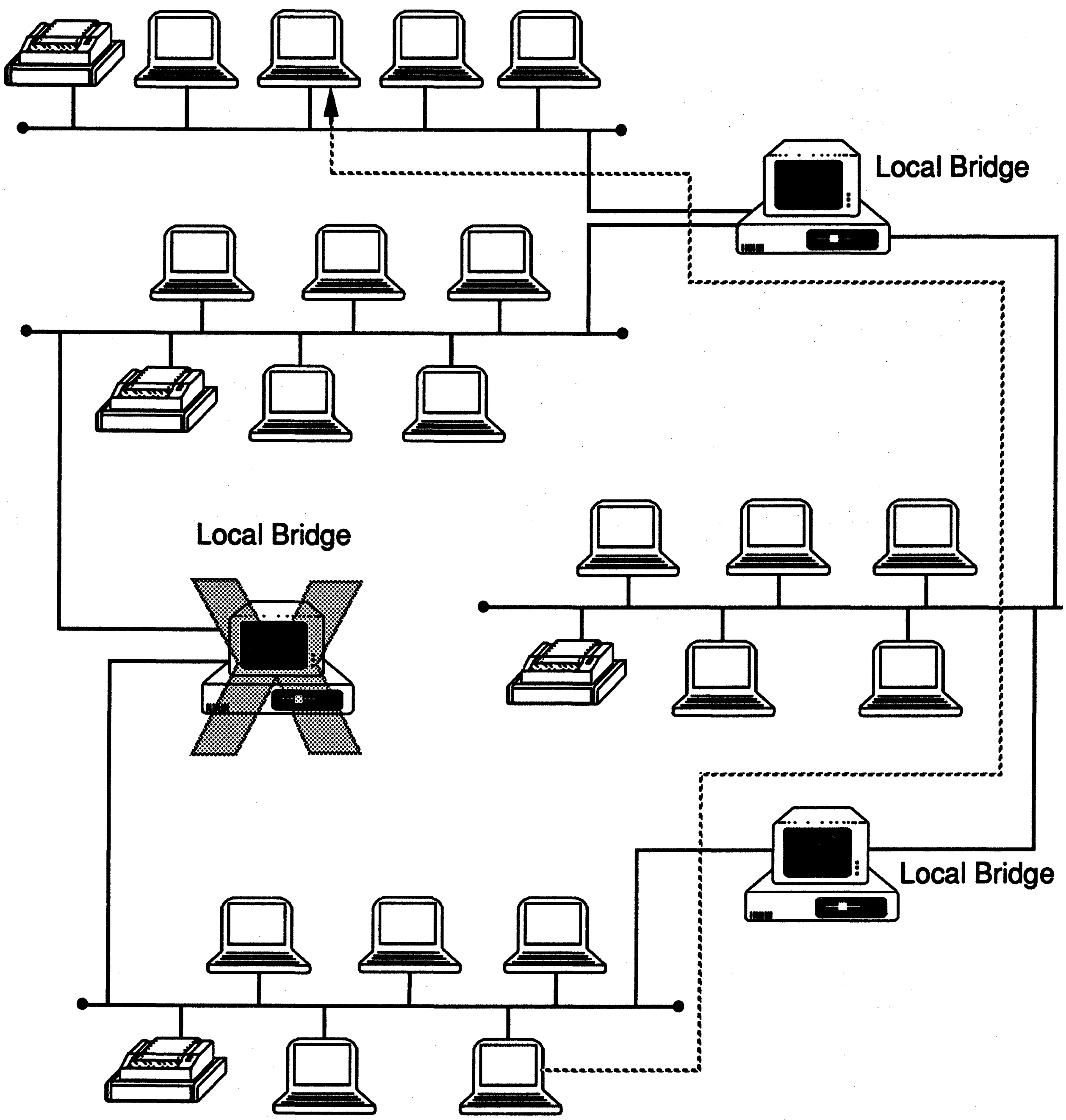
If the bridge does not have the destination address in its address table, it assumes it is foreign and forwards it. This leads to an interesting question. In Figure 6, we have three LAN segments connected with a single bridge. If the bridge gets a frame from segment A with a foreign address, where does the bridge send the frame, to B or to C? The answer is both. The bridge makes a copy of the frame and forwards one copy to each of the possible locations, a process known as flooding. If the address is valid at all, the receiving station will answer. When the answer frame passes through the bridge, the bridge will discover the new address' location and can add it to the address table.

Since a bridge links LANs into a single, large LAN, all the addresses anywhere in the new internetworked LAN must be unique. You can understand how this would work by analogy to street addresses. If we all live in Tallahassee, there can only be one 129 Main Street. In fact, LAN managers encounter many problems stemming from addressing conflicts in which two or more LAN citizens have the same address, or where a station repeatedly tries sending a message to a nonexistent address. In the first case, the sender can elicit an "answer" from the wrong device and in the latter case, the LAN is flooded with nonsense packets.

Servers as Bridges

Since a server is the primary enabler of most services on PC LANs, servers have been critical elements in developing effective LAN internetworks. Internetwork servers function in two ways: as internet devices themselves (bridges, routers, gateways) and as the major vehicle for

Figure 4.
Spanning Tree



Spanning Tree prevents loops by choosing a single route and disabling all other potential routes.

delivering information-sharing—the major reason for creating the internet in the first place. When end users anywhere in the internet can access any information on any server, your company's effectiveness is multiplied.

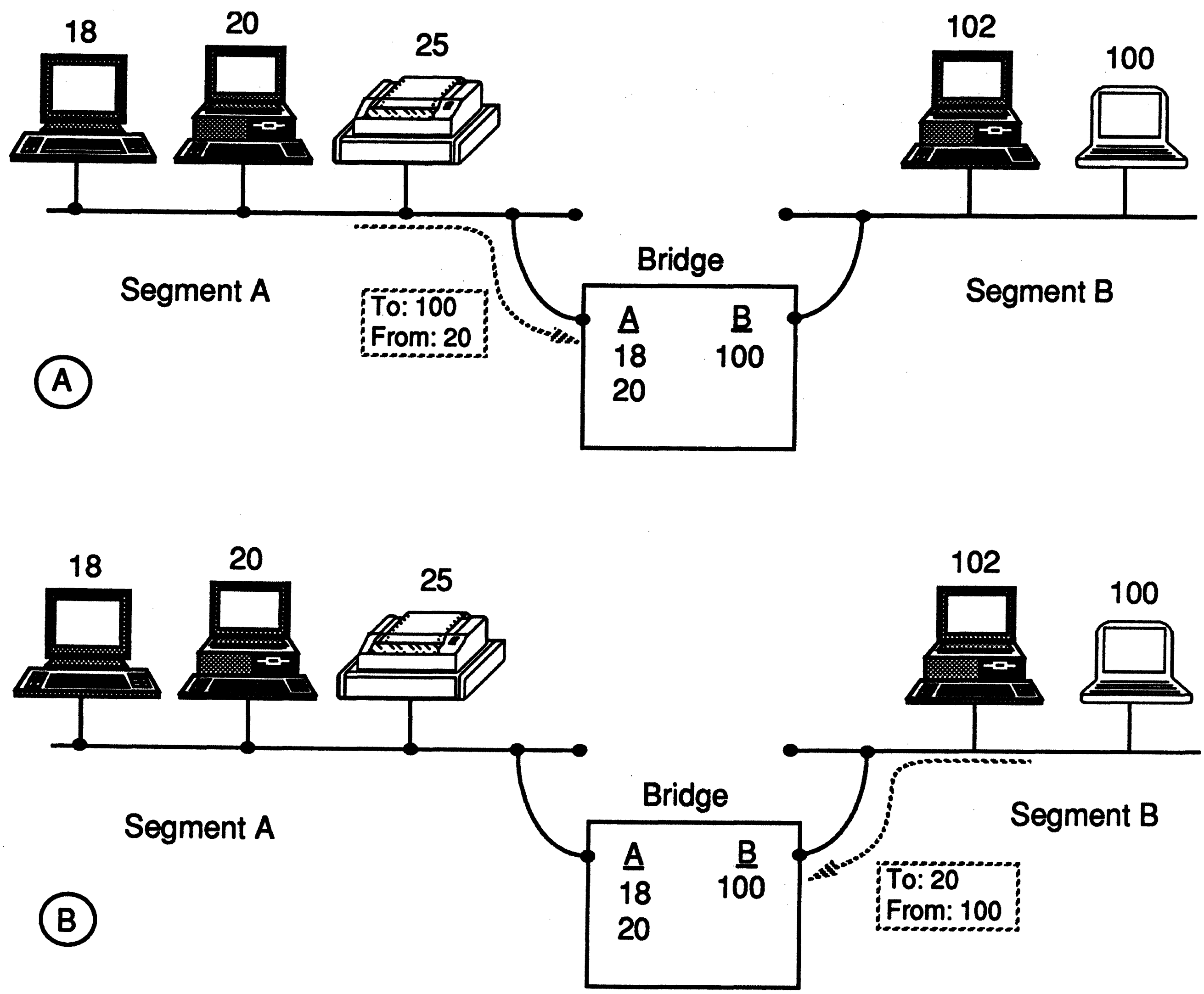
Managing a Bridged Internet

Bridges are an excellent choice for simply extending the size of a single LAN. From a management perspective, since bridges are only yes/no pass-through devices, a bridged internet is a single-management domain.

A bridge can be used as a management tool with programmable filters. It can, for example, be programmed to reject all incoming broadcast frames. This type of segment is not troubled by sometimes wasteful broadcast traffic. A segment that is involved in extremely secret development work is sometimes isolated in another way, with no outgoing traffic allowed. Smart bridges can be programmed to filter in any way that involves data link information: size of the frame; network layer protocol (contained in Other Data Link Layer information in Figure 6); specific addresses, and so on. We'll discuss smart bridges again later in this report.

When considering whether to buy bridges for internetworking, you must decide what traffic level you can expect

Figure 5.
Learning Bridges



A learning bridge generates device address tables automatically.

on your LAN and the types of devices that you expect to be LAN citizens. If the incoming traffic is more than the bridge can process (bridges have relatively small buffers), it will discard the overload frames. If so few consecutive frames are forwarded that most applications are affected, the bridge is said to be choking on the traffic. High-bandwidth applications like CAD/CAM, desktop video, simulations, and other graphics-intensive users can easily choke bridges.

Ethernet to Token-Ring Bridging

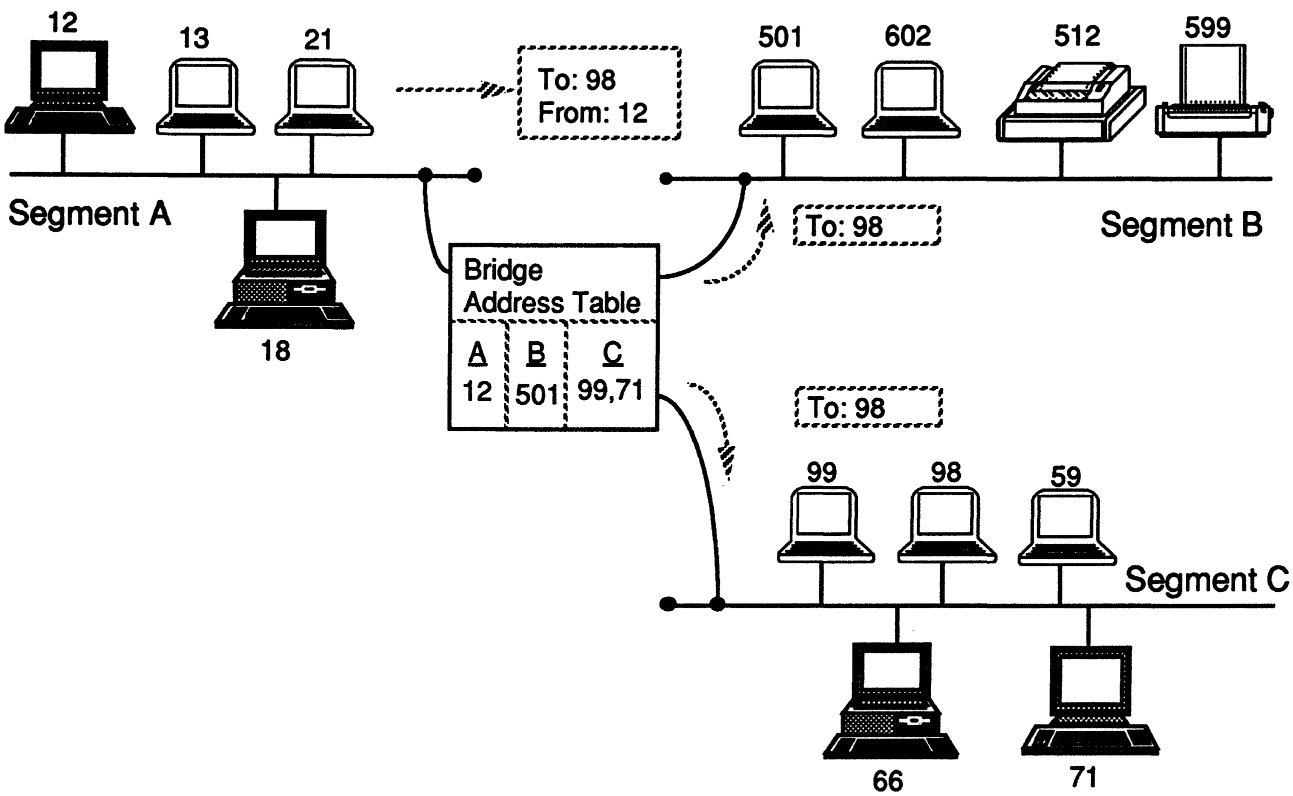
Note: Token-Ring is an IBM product; token-ring is a generic technology specified by 802.5.

Ethernet and IEEE 802.3-compatible LANs have the largest installed base, while IBM and 802.5-compatible token-ring networks have taken a narrow lead in terms of new shipments. Therefore, Ethernet bridges outsell all other bridge types, closely followed by IBM Token-Ring bridges. The newest frontier in bridging is the .3/.5, or Ethernet/Token-Ring, bridge.

The .3/.5 ("three-to-five") bridge poses a considerable technical challenge to designers for three reasons:

1. The difference in access method. Ethernet's CSMA/CD creates a contention environment while token-ring insists on deterministic access. Clearly, the latter kind of bridge must wait for a token before it can effectively forward Ethernet packets and must therefore have a larger-than-average buffer. It will be less than completely transparent from the Ethernet user's perspective and not quite deterministic from the token-ring user's perspective.
2. A Big Pipe/Little Pipe issue in which a 10M bps Ethernet is feeding into either a 4M bps or a 16M bps IBM Token-Ring product, and up to 80M bps with other token-ring vendors. (The 100M bps FDDI networks really exacerbate the problem, unless the FDDI ring functions only as a backbone.)
3. Finally, and most challenging, 802.5 networks have an aberrant method for creating a connection between sender and receiver. In contrast to the 802.1 Section D

Figure 6.
Flooding



A bridge floods when it receives unknown addresses.

Spanning Tree specification, which governs all other standard LAN types and in which the internet devices create the path, 802.5 networks require the end-user device to create the path, a process called source routing. (In this discussion, ignore the fact that a bridge is doing something called routing.)

The disadvantages of source routing include the heavy burden it puts on user devices, which may not have memory available for route-building functions, and the fact that the source routing algorithm is not as robust as Spanning Tree. Three-to-five bridges obviously need to map these two loop-prevention methods to each other, as efficiently as possible. The IEEE's new Source Routing-Transparent Bridging (SR-TB) standard specifies how the bridge arranges for source route bridging on the 802.5 side and spanning tree bridging on the 802.3 side. Most vendors producing 3-to-5 bridges have announced support for this and related standards. Some non-IBM token-ring products follow the 802.1D (spanning tree) specification, so these bridges cannot interoperate with source routing token-ring bridges.

Routing (Layer 3)

The purpose of a router corresponds generally with the OSI Model's Layer 3 functions: efficient traffic flow and management of end-to-end packet flow. A router must be able to route traffic using both datagrams and virtual circuits.

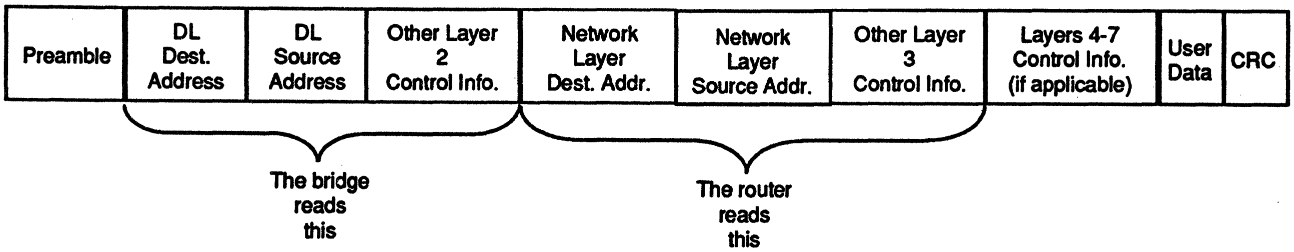
Routers as Traffic Managers

In order to make its routing decisions, a router reads the packet's network address, which is contained in the third layer's control information (see Figure 7). Most LAN users think of their workstation address as a simple string of numbers, corresponding to the address of the network interface card; for example, 12345A2. The full workstation address is actually 239/12345A2, where 239 is the network address. The full device address is equivalent to 129 Main Street, Topeka, Kansas. If we all live in Topeka—or are citizens of the same LAN—we can state our address as 129 Main Street. To communicate with someone outside of Topeka—or network 239—we must specify the full address. This example points out another way of thinking about routers: a way to set up a connection to another, separate LAN, to get "off net." In an internetwork using routers, you are linking separate networks with distinct network IDs. Each individual LAN may use bridges to make connections between its segments. In this type of internet, the individual LANs are subnets of the internet.

Because a router reads the network layer address, it is oblivious to the addressing and other issues of the Data Link layer, but it is protocol specific. This protocol-sensitivity makes the router singularly unlike the bridge, which will pass any LLC-compatible frame, whether it belongs to Digital's DECnet, IP, XNS, or another stack. When deciding when and how to use routers in your network, remember that a router can only manage traffic that belongs to a specific protocol stack.

This protocol-specific nature can create problems for network managers who must manage internetworks using many different protocol stacks, if each individual stack

Figure 7.
Bridge and Router Packets



Bridges and routers read different parts of a typical packet.

needed its own separate set of routers. Fortunately, multiprotocol routers exist, usually composed of a hardware platform with management and protocol-filtering software on one or more individual protocol boards. After deteming the incoming packet's protocol identity, the filtering software passes the packet to the appropriate protocol board.

It Pays to Advertise

A router functions by providing services when requested, but it takes an additional step—it advertises as well. A network station that wants to send one or more packets off-net sends its request directly to a router, addressing it specifically. A router listens for its address and processes only packets addressed to it, unlike the bridge which processes every packet that it encounters. If the incoming packet is a single (datagram) packet, the router forwards it individually. If the incoming packet is a virtual circuit setup request packet, the router will decide whether a virtual circuit is feasible; if it is, the router assigns a virtual circuit number and informs the sender to proceed. Meanwhile, the router is communicating directly with either the destination station or with the next router in the path. While the circuit is active, the router is the immediate destination of all sender-generated packets and it provides flow control on the circuit.

The end-user stations know the router's address because every internet router transmits an "advertisement" broadcast packet at regular intervals, informing the citizenry that it is in business and providing its addresses. That is also how the routers learn of each others' existence, allowing them to build directories of routers and the network addresses to which each can build a path. These directories are known as routing tables. The internet's routers communicate with each other regularly concerning their traffic load levels and the networks to which they can make a connection, as well as the cost associated with each route. This "cost" can be measured in the number of intervening routers between sender and receiver, called a hop count. The cost can also be measured in milliseconds-of-delay or in other factors. Routers use routing protocols to communicate with each other, and it is these routing protocols that create routing tables and calculate costs. The Routing Information Protocol (RIP) counts hops, while "Hello" measures delay, for example.

Consider a user station on network 100 that must make a connection to a printer on network 230. The user station sends a request to its local router, which knows about another router that, according to its routing table, knows how to get a packet to network 230. Your local router's routing table allows it to build a route to any network in your internet. We say that routers "know the entire network's topology." (This isn't technically correct when you are using one of the newer protocols that segment information flow, but it is absolutely correct in a functional sense.)

Routers learn about local addresses on their own networks by listening to the source addresses of locally-addressed frames. In some routing protocols, end user devices "advertise" directly to their local routers.

Using Routers

Routers must find the best route between sender and receiver. A network manager determines what constitutes "best" by choosing a routing protocol and programming optional parameters, if any. A router can choose the cheapest WAN link, or alternately, choose the fastest. In either case, the router must try to balance network traffic loads by choosing the least busy of all available routes, a process known as load balancing. In order to make the best use of alternate routing, of course, there must be more than one possible way of getting from sender to receiver; therefore, an internet generally has two or more routers per network.

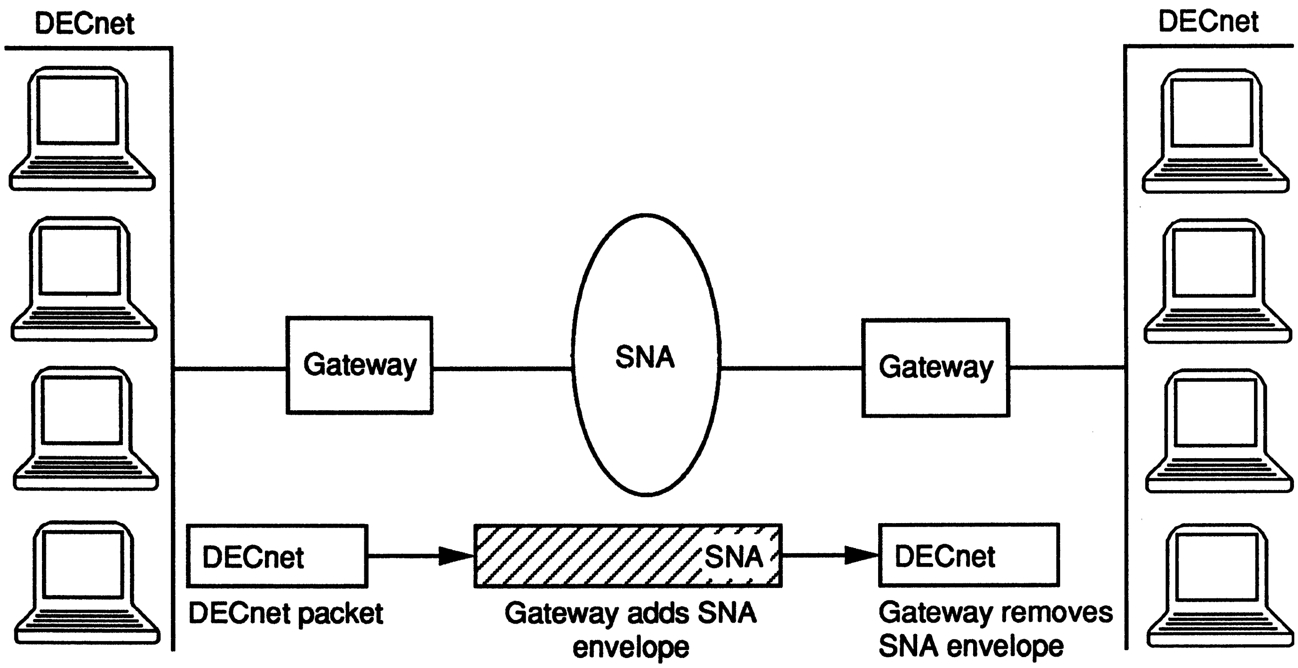
In some cases, the choice of routing protocol has a profound impact on your allowable topology. Using RIP, an internet packet has a maximum allowable hop count of 16, after which it is discarded as too old. If you are designing very large RIP internets, two networks that are more than 16 hops away from each other will be unable to communicate because they will never learn of each other's existance. A remote connection generates two hops so RIP is especially problematical for WANs. Similarly, if you are using the Hello routing protocol, two networks may be effectively cut off from each other if the accumulated delay across the intervening internet devices exceeds the maximum delay. Newer routing protocols, notably TCP/IP's Open Shortest Path First (OSPF) and OSI's Intermediate System-to-Internediate System (IS/IS) were specifically designed to handle the needs of large, very complex internets, and most vendors have announced plans to support one or both of these standards.

New Router Functions

In addition to simply balancing traffic loads, some routers can be programmed to give certain traffic types priority when bandwidth is limited—a process called priority routing. For instance, IP packets can receive priority over IDP packets, or graphics workstation traffic can be routed in preference to other traffic requests.

A new development still being tested in universities and research laboratories is the concept of bandwidth reservation, whereby a connection setup propagates all the way to the end station before the sender gets the go-ahead to

Figure 8.
Gateway Tunneling



Two DECnet LANs using gateways tunnel across an SNA network.

transmit. This system developed in response to the problem that many high-bandwidth applications have encountered in already loaded networks, whereby the first router provides bandwidth but the connection is blocked later in the setup. Meanwhile, the first router has surrendered a significant amount of its available bandwidth (fruitlessly, as it turns out), blocking other users, while the connection will not succeed because of bandwidth blockage later in the circuit. If the entire connection were given approval only after all bandwidth were allocated, high-bandwidth applications would connect more frequently and other users would be blocked less often, making routers more efficient. This new development presents many thorny technical problems, but research is continuing because of the incredible proliferation of graphics-intensive and other high-bandwidth desktop devices since 1986. The move toward high-bandwidth user devices is one of the major trends fueling the transition from bridges to routers as internetworking devices.

Routers as Management Devices

Routers can also perform other essential management functions in your internet. For instance, since the router segments an internet into separate networks, problem determination is faster—looking for the problem in a single, smaller LAN yields a smaller number of potential culprits. In addition, a router can function as a network "firewall," isolating failures and problems to a single LAN. Bridges, because they do their decision-making almost exclusively on the basis of data link addresses, allow many damaging packets to proliferate throughout the LAN. A router firewall can keep problems local. When you consider the number of network problems that can cause "Ethernet meltdown," firewalls become attractive. Therefore, segmenting your network with routers gives you added error management as well as traffic management.

Routers have their disadvantages, of course. For example, routers generally cost more than bridges because of their more sophisticated functions. Routers are also protocol-specific, so each protocol must be present in all the routers in order for the internet to function. Routers within each protocol group must also use the same routing protocol. Unless you choose completely CCITT/ISO standards-based products (and sometimes even then), vendor implementations of protocols vary enough to make it advisable to buy all your routers from a single vendor. This constrains your choices in both initial installations and in migration to future needs.

Gateways

A gateway does more than simply read a packet's address; it actually restructures the packet in some way. If a DECnet packet must be passed to an SNA network as an SNA packet, it must pass through a gateway. In effect, a gateway is a protocol translator. Because the gateway must read and then re-create the packet headers in a new form, it always affects performance to some extent. The newest and most cost-effective—in dollars and performance—form of gateway is one that allows packets to "tunnel" through a foreign network architecture. Instead of changing the packet's headers to conform to the new architecture, an architecture gateway can simply put the complete packet into an envelope of the new protocol stack. This re-enveloping process can occur much more quickly than traditional gateway processing. In Figure 8, two DECnet LANs using gateways tunnel across an SNA network. (Protocol names are used here only for illustrative purposes. Any protocol with the appropriate gateway can tunnel in this manner.)

Layer Seven Gateways

A gateway functioning at the Application Layer is usually an application gateway. The most popular type of application gateway is the electronic mail gateway, based on the tremendous popularity of electronic mail systems. Most LAN architectures include two or more electronic mail programs; some application gateways translate from one proprietary package to another, while others translate to OSI's mail protocols in the X.400 family. Currently, X.400 software exists in two forms: as the transport method between two incompatible proprietary mail systems, and "full native implementation," which means that the two communicating systems generate their messages directly in X.400 formats.

Multiprotocol Internetworking

An architecture gateway that re-envelopes packets can also allow communication between incompatible devices on the same physical LAN. Two protocols can coexist on the same LAN, logically distinct from each other. An architecture gateway can act as an intermediary, translating between two formats and allowing the two logical networks to become one.

You can also bring different logical networks together through the use of a multiprotocol server. Multiple protocol stacks on the same server allow users to access the same databases through their individual protocol environments. A server that allows simultaneous access through OSI, TCP/IP, DECnet, and XNS stacks, for example, reduces your dependence on an architecture gateway.

Layer Three (X.25) Gateways

Network Layer or X.25 gateways reformulate LAN packets into packets suitable for traveling on a public or private wide-area packet-switching network. With a well-designed internetwork, several departments, workgroups, or floors, each with their own LANs, can use the same gateway and have a single interface to the outside resources. With this single connection to public databases, the entire internet funnels its requests through a limited number of devices, significantly lowering costs compared to allowing each individual user to make a separate connection. Disadvantages of network-based gateways include the loss of all connections if the single device fails or if the network is down for any reason. To forestall this potential danger, many companies use hot standby gateways or gateways running in parallel.

Users can install an internet gateway board on an existing LAN server. A communications server is a better candidate for an X.25 gateway (if the internet has one) than a general-purpose server, because a gateway will create a heavy work load for a server. Many internets that experience moderate to heavy wide-area communications needs install a dedicated X.25 gateway server.

Hybrid Products

Some of the most exciting products on the market today are hybrid products, embodying some functions of two or more separate products in a single box. Some smart bridges can deliver minimal least-cost routing over WAN links and even some minimal load balancing. Managers choose smart bridges for growing internets that do not need router-to-router communication, sophisticated load balancing, or parallel operation.

Both terminal servers and X.25 gateways are popular choices for combination with a sophisticated bridge. These server/gateway combinations are generally created by adding a new functional board-level product to an already existing product line.

Managers can also choose true multifunction boxes containing both bridges and routers. These integrated bridge/router products include a manager board to read the incoming packet's network address and determine which protocol stack it belongs to. If it recognizes the protocol, the manager board passes the packet on to the correct router board. If it does not recognize the protocol, it passes the packet to the bridge board, which performs data link filtering and forwarding. In this way, bridging is the function to which an unknown packet defers, allowing the LAN to perform some kind of processing even on the most foreign packets and removing the major disadvantage of the router-only internet.

Smart bridges and routers are also being integrated, as boards or as software, onto other mainstays of the data communications infrastructure, especially:

- Packet switches and packet assembler/disassemblers (PADs)
- Frame relay switches
- T1/T3 multiplexers

These integrated devices are truly revolutionary, allowing network managers to integrate functions into a smaller footprint device, saving space and management time. They also allow network managers to potentially manage more components of their infrastructure from a smaller number of access points. Bringing the entire infrastructure under a single management umbrella is the major theme of the management section of this report. In addition, the addition of board-level routers into T1 networks (as Network Equipment Technologies did recently with its LWX router) allows T1 users to transition to routed internets without the addition of a complete new device and management structure. The addition of frame relay options to existing integrated bridge/routers is another exciting development, since frame relay is a very cost-effective Data Link Layer wide area connection protocol.

LAN Internet Management

As the company's LAN grows larger and larger, the hidden costs of inefficient (or nonexistent) LAN management grow exponentially. An internetwork is an intrinsically more challenging management task than a single departmental LAN. As the number of devices and linkage complexity increases, so does the possibility of protocol mismaps and other more sophisticated misunderstandings. In addition, a larger number of devices leads to a larger number of failures, which are increasingly more difficult to pinpoint; errors also proliferate their effects across a wider area.

Toolkits

Protocol analyzers, especially the portable type, help leverage human resources and in fact supply information not obtainable from any other source. The cost of proper tools—even the most sophisticated and expensive—represents a small fraction of the dollar value of your network management staff's time. When a network fails, the cost of tools becomes an even smaller percentage of the value of lost productivity. For a company to consider the network a corporate utility, and an essential weapon in the battle for market share in its industry, the network must achieve both high reliability and high bandwidth utilization. Perhaps most importantly, the more reliable the network, the easier it will be to cost-justify the next new network purchases to upper management.

Standards

Since every vendor of LAN, WAN, and general communications hardware and software sells its own network management package, the challenge becomes integrating this array of incompatible systems. Most managers settle for multiple network management consoles, but a single network management standard is the ultimate user need.

A network management standard requires that all devices in the internet report status, traffic levels, and other information into a common database, in a common format, for later playback in tabular or graphic form. The OSI Network Management standards stipulate a three-tiered network management architecture which ends in a network management console from which a manager can monitor and control any internet device. Both AT&T and Digital Equipment Corp. have announced product architectures that conform to this architecture specification—AT&T's Unified Network Management Architecture (UNMA) and Digital's Enterprise Management Architecture (EMA).

OSI's approach to network management is referred to as layer management, because each layer includes management functions. In addition, the Application Layer's Common Management Interface Services (CMIS) specification has spawned the Common Management Interface Protocol (CMIP), on which many internet and LAN vendors are anchoring their network management architectures. The related Common Management over Logical Link Control (CMOL) specification, just released, will allow managers to manage networks that carry applications and logical link control, without any of the intervening layers' functionality and software.

The TCP/IP community created a simpler standard, called the Simple Network Management Protocol (SNMP), for managing bridges and routers in a typical internet. SNMP has been endorsed by most internet device vendors as an infrastructure-management standard.

Security in Your LAN Internet

Internetworking certainly leads to more information sharing. As the internet manager, you must ask yourself: Am I doing more information sharing than I know about? As network complexity increases, an intruder has access to more LANs and more LAN databases in the course of a single successful break-in. In this sense, isolated LANs are safer LANs.

Physical Security

The first level of network security is physical security, including secure access devices and a secure medium. Access devices are generally user PCs and workstations. As a first line of defense, every user should be required to use a long (12 characters or more) password that does not include names of children, spouse, pets, or favorite sports team. Passwords must be changed at regular intervals to be effective. Even ancient Hittite was dicipherable when philologists had a large enough sample to work with! Your task is to make sure that an intruder doesn't have that chance with your network users' passwords.

When the access device is a dial-in device, your network is especially vulnerable. Many network managers disable the dial-in ports after business hours; this causes minor inconvenience to legitimate users and maximum inconvenience to intruders. Call-back modems, which receive logon requests and then dial back to the user's authorized telephone number, are very popular as well, because they can effectively block all unauthorized users. Since they transfer the telephone charges to the network owner, users should be employees, not customers, unless there is an effective charge-back method.

To protect the medium itself, you can choose intrinsically tap-resistant media like fiber optics, or you can try to secure the medium within walls, as managers do with existing twisted-pair wiring converted to LAN use. Many network managers value the star topology with unshielded twisted-pair wiring simply because the star is associated with wiring closets and closets can be locked. On the downside, if an intruder is an insider with access to wiring closets, concentrating the medium in one place provides easier access to all the LANs in one place.

What an intruder does with access to the network medium is called packet sniffing, a colorful metaphor for the process of copying packets—including user data—from the data path. The tools that intruders use for security breaches are the same extremely sophisticated tools purchased for network management, making them a double-edged sword in the hands of your network management staff.

Software Tools

Government-sponsored research, especially military projects, frequently require that network data be encrypted; that is, scrambled to make it unreadable without access to an unscrambling key. You can encrypt the entire packet or you can encrypt only the user data. Each approach has its drawback in a complex internetwork. Encrypting only the user data (end-to-end encryption) leaves the addressing in the clear so bridges and routers can read it easily. This means the encryption will not cause performance bottlenecks, but it also means that an intruder will be able to trace traffic patterns—extremely useful information in an internet. Data Link Encryption (DLE) encrypts the entire packet, so even traffic patterns are masked. Since the bridges and routers must decrypt each frame and packet before processing it, however, users may experience delays. In addition, the packets are vulnerable to attack at the internet devices themselves when they have been decrypted. If your internet devices are not physically secure, all the costs of encryption may be wasted.

Moving beyond the physical medium, you can take precautions to protect your databases—servers, generally—by locking them away from user access in network management centers. Most LAN vendors have concentrated on making their servers easy to use, and as a result have created systems that are security nightmares. The network operating system must engage in self-protection at the file level by performing authentication checks for every user on every nontrivial file. This means that before users are given access to most files, they should be checked against an authorized list of users.

Authentication Standards

National and international standards bodies have not ignored the security issue, but it has taken a back seat to medium, transport, and applications issues. Authentication is primarily a Presentation Layer (OSI Layer 6) process.

On the national level, the IEEE's 802.10 subcommittee is developing a LAN security standard, while ANSI has developed several LAN and internet security proposals. Clearly, as users become more aware of the need for security—and the need to pay the necessary premium for a secure LAN—work on authentication and other security standards will receive the attention it requires. The ultimate challenge is to create networks that are easily used by legitimate users doing authorized things, while functioning as impregnable fortresses to all other potential users.

Looking to the Future: Intelligent LAN Internets

The internetworks of the 1990s will be larger and more complex, both from a technical and an applications sense, than ever before. These new internetworks will tax the resources, persuasiveness, and creativity of network managers as they tackle both day-to-day and long-term challenges.

Device Trends

LAN vendors will migrate as much intelligence as possible into internetwork devices of the 1990s, reducing the load on user devices and on the people who manage them. Routers with greater intelligence and more sophisticated routing protocols will lighten the network management load on network administrators. Network managers will continue to choose standards-based products, for both lower management costs and higher reliability in complex internetworks.

The mix of internetwork devices in a typical installation will include more multifunction devices, with small footprints and as much individual functionality as possible moved into board-level products that allow a mix-and-match to fit user needs. The simple Layer 2 device has become a commodity, judged primarily on speed and price, while smart bridges will accelerate their integration into routers. Both bridges and routers will continue their integration into the WAN infrastructure, becoming integral parts of packet switches, frame relay nodes, and T1 multiplexers. Metropolitan network structures will become more common, giving companies an intermediate option for medium-distance LAN-to-LAN connections. More innovative products that satisfy user needs for standards compatibility, modularity, and high performance will continue to be best-sellers.

By 1994, the internetworking industry will cease to be a segment of the LAN market and will have migrated to include many WAN and MAN technologies. Internetworking companies will be acquired by LAN companies, as the latter are acquired by computer and communications companies.

Ultimately, LANs must be able to internetwork not only with each other but also with the emerging WAN infrastructures: CCITT/ISO's integrated services digital networks (ISDN), frame relay, Bellcore's synchronous optical network (SONET), and switched multimegabit network services (SMDS). Although ISDN's 64K bps data channels are significantly smaller pipes compared to 10M bps LAN speeds, LANs will need to internetwork with voice, data, and video transmissions of local and nation-wide ISDN pipes. IEEE subcommittees and the ISO/CCITT joint technical committees have developed several such interface standards to link LANs to ISDN.

SONET has gained some success as a competitor to FDDI as a LAN backbone technology as well as a WAN option for telcos and interexchange carriers, but it has not been a market success (yet) outside of the telco industry. SMDS has been generating a significant amount of interest as a high-bandwidth technology but is still emerging, is not established, and has the disadvantage of being a purely North American phenomenon while its competitor—ISDN—has international backing. Both SONET and SMDS can deliver gigabit data rates. It remains to be seen which of these technology options will win in the marketplace. In the short term, most internetworking vendors plan to support one or more of them. FDDI, ISDN, and SMDS have been designed to (theoretically) allow interoperability, so users can expect migration routes to exist from today's choices to tomorrow's options.

Management Trends

An internet needs proactive management to complement problem determination and resolution, defined as reactive management. Managers must use the best available tools to ascertain and plan for future user needs. In order to do this, they will need facts and figures on network effectiveness to convince CEOs and other nontechnical management that investing in the network is a surer way to increase profits than sending the top sales producers to Hawaii for a week. A really effective internet manager will be someone who can convince management to not only invest in an internet upgrade but also send the network management team to Hawaii.

New Technologies

As if today's unsolved problems were not enough challenge, network managers can look forward to some additional challenges in the future. More high-bandwidth applications running on more and more high-powered workstations will press today's 4M and 10M bps networks to their capacity limits and beyond. These applications will easily choke even the fastest bridges, affecting internet access for other users. Network managers need bridges with flow-control capacities (so-called choke packets, now available only in laboratory LANs), faster throughput, and—especially—the capacity to allocate bandwidth on all intervening internet devices before a virtual circuit is established.

Emerging network technologies and standards will further compound LAN big-pipe/little-pipe problems when connecting both to ISDN and to the fastest wide-area infrastructures. Even T3 capacity at 45M bps cannot match a local network operating at FDDI's 100M bps.

Finally, the internet must assume the majority of the intelligence for day-to-day activities through intelligent routing, more effective network management, and the migration of artificial intelligence (AI) software to internet devices. Corporations must reserve their network management personnel for proactive management tasks that will make the internet a genuine tool for the company's success in its marketplace, while migrating as much of the fire-fighting intelligence as possible from people to software. ■

This Datapro report was written by Victoria Marney-Petix. Ms. Marney-Petix designs and delivers on-site seminars on internets and network management and consults with vendors on new product development and competitive strategies from her office in Fremont, CA. She is the author of three networking books, most recently *Client/Server Computing* (1991) and *Mastering Internetworking* (1992).

Implementing CD-ROM Applications on a LAN

In this report:

Datapro Summary

Prior to implementation on a LAN, users must understand how a CD-ROM application functions. Factors specific to CD-ROM applications include CD-ROM hardware (drives and controller cards), CD-ROM device drivers, Terminate and Stay Resident (TSR) software known as Microsoft Extensions, RAM CRAM issues, and "spill files." Two general configurations, decentralized and centralized, can be used to implement shared, multiuser access to CD-ROM drives on a LAN. Many organizations already have LANs with file servers, modem pools, gateways, and other devices. Shared, multiuser access to CD-ROM applications requires a CD-ROM server. Users must plan for a CD-ROM server by considering flexibility and future expansion issues. It is also important to understand how a CD-ROM application must be configured to operate properly on a LAN. CD-ROM applications that span multidisk sets, such as Medline or the Census Data (which requires 10 or more contiguous CD-ROM disks), can create a real integration challenge. Market leaders are needed to produce useful third-party utility programs to simplify the integration of CD-ROM databases on a LAN.

CD-ROM Applications—Pre-Networking

Before addressing how CD-ROM applications operate on a LAN, it is important to understand how CD-ROM applications have been developed. Keep in mind that there are hundreds of different CD-ROM developers, many of which envision the market as having standalone CD-ROM workstations, not networks. Most CD-ROM developers have not yet come to grips with LAN issues.

To better understand CD-ROM applications, it is helpful to understand the components of a CD-ROM application.

CD-ROM Application Components

CD-ROM Disk as a Data Source: One great benefit of the CD-ROM disk is that it can store over 600 million bytes of information. Generally, this data is comprised of full-text information and the indices to the full text. More recently, images, audio, and video have also been placed on CD-ROM disks.

Search/Retrieval Software: The application developer almost always provides proprietary search/retrieval software. A full 95% of the applications use MS-DOS based software, usually supplied on diskettes but sometimes available on the CD-ROM disk itself.

Spill Files: When a CD-ROM application is loaded into memory, the application creates temporary spill files. The CD-ROM application generally attempts to create these spill files on the C: drive of the hard disk.

—By Howard McQueen
CD Consultants, Inc.

For a CD-ROM application to run in a standalone PC, the PC must have additional hardware—a CD-ROM drive, CD-ROM controller card with cable, and CD-ROM device driver software. The CD-ROM controller card is inserted in a free slot in the PC and a cable is used to connect the controller card to the CD-ROM drive.

Since each CD-ROM drive manufacturer's equipment is different, a CD-ROM device driver (a software file) is supplied with each CD-ROM drive so that the PC is able to address the specific drive. There is an installation process which installs the device driver on the hard disk of the PC, then modifies the CONFIG.SYS file (to call the device driver).

Spill Files

Once running, the CD-ROM application creates spill files to increase performance. Spill files are created when a user starts an extensive search. The application will "spill" preliminary search results to disk and use the hard disk to complete the search process. Once the user exits the application, the spill files are removed. Applications generally default to creating spill files on the C: drive. Spill files cannot be created on the CD-ROM disk because it is read only memory (ROM).

RAM-Hungry CD-ROM Applications

CD-ROM developers created their applications based upon having 640KB of RAM available in stand-alone workstations. After loading MS-DOS, the CD-ROM device driver and MSCDEX (Microsoft CD-ROM Extensions), the memory remaining in the PC is generally more than 540KB. Many CD-ROM developers designed their applications to use up to 540KB, not having the foresight to recognize the memory demands that LAN connectivity software imposes.

Microsoft CD-ROM Extensions (MSCDEX)

Standalone CD-ROM applications require Microsoft Extensions for two fundamental functions. The first function is mandatory and the second is optional.

Primary (Mandatory) Function of MSCDEX: Microsoft developed MSCDEX to allow MS-DOS to be able to read a CD-ROM disk. Since the CD-ROM disk (600MB storage capacity) exceeded the early MS-DOS volume storage limitation of 32MB, the extensions allowed MS-DOS to look at the CD-ROM disk as a large read-only hard disk. It is for this reason that any stand-alone PC with a CD-ROM drive requires MSCDEX to be loaded into memory. Without MSCDEX, a PC cannot read the disc in a CD-ROM drive.

Secondary (Optional) MSCDEX Function: MSCDEX is also an Application Programming Interface (API) which allows developers of CD-ROM applications to make calls to MSCDEX to retrieve data and other information off the CD-ROM disk. It is not mandatory that developers use MSCDEX. CD-ROM applications can be developed with normal DOS programming calls. Today, a small group of important CD-ROM applications still use MSCDEX's API functions.

Now that the components and environment unique to CD-ROM have been reviewed, the focus of this report will turn to the issues involved in implementing CD-ROM applications on a LAN.

Decentralized versus Centralized Configuration

Decentralized Configuration

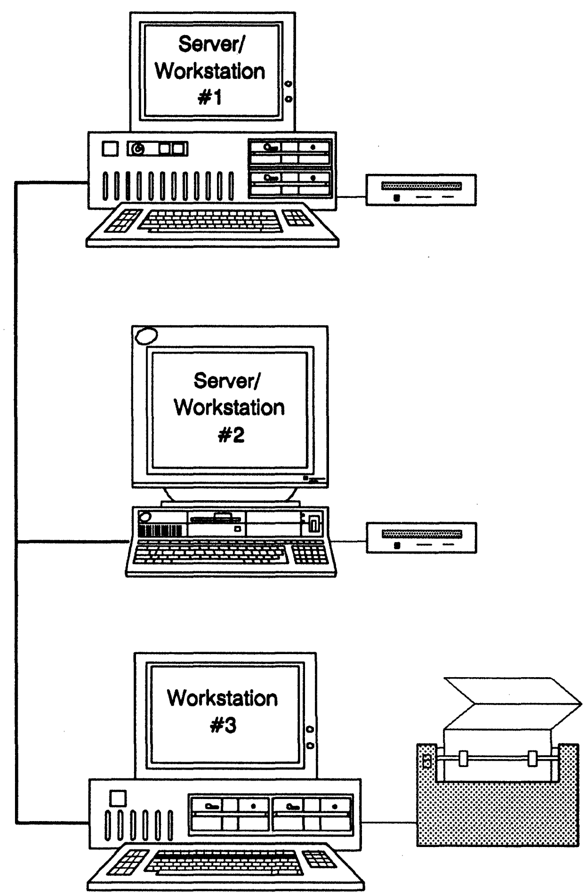
If an organization owns a number of PCs configured with CD-ROM drives, a configuration such as that shown in Figure 1 can be considered for sharing CD-ROMs over the LAN.

The decentralized configuration appears ideal for utilizing existing equipment and allows sharing of CD-ROM drives in any workstation connected to the LAN. An example of a commercial network operating system (NOS) that provides support for this configuration is LANtastic, by Artisoft.

LANtastic comes with built-in CD-ROM sharing support. Any workstation with a CD-ROM drive can be shared over the LAN. The PCs that have CD-ROM drives can become shared servers and, at the same time, remain workstations. Any NOS which can provide this dual workstation/server capability creates a peer-to-peer LAN environment.

Another benefit of this configuration is that only workstations sharing CD-ROM drives are required to load MSCDEX (workstations #1 and #2). Other workstations (i.e., workstation #3) that access the CD-ROM drives via

Figure 1.
Decentralized (Peer-to-Peer) CD-ROM LAN Configuration



A decentralized configuration may be an effective method to use existing equipment and allow sharing of CD-ROM drives by any LAN workstation.

the LAN do not need to load MSCDEX. This saves a minimum of 27KB of memory in each workstation without CD-ROM drives.

The peer-to-peer LAN configuration is the least expensive way to create shared CD-ROM access. Cost alone, however, should not always be the determining factor in implementing a network design. Equally important factors to consider are reliability and performance.

Limitation

A peer-to-peer configuration allows an existing workstation (with CD-ROM drive) to continue to be used as a workstation and also take on the role of CD-ROM server. In Figure 1, assume that all three workstations are accessing the CD-ROM drive attached to workstation #1. What happens if:

- The user of workstation #1 reboots that machine?
- The application running in workstation #1 hangs?
- Workstation #1 encounters a hardware parity error?

In all of the above conditions, workstation #1 ceases to function as a workstation and also ceases to function as a shared CD-ROM server. The author is always reluctant to implement a system design which allows a single point of failure to adversely affect other users on the LAN. This is the major weakness of peer-to-peer networking.

Even if the environment is relatively free of the faults just discussed, anytime a MS-DOS based PC is configured to perform two processes concurrently (workstation and server), it will perform both functions, but in a greatly reduced capacity. Response time will suffer as more users share the CD-ROM.

Earlier in this report, the two roles of MSCDEX were reviewed. The second role, that of an API, is still used with some CD-ROM products. LANtastic's built-in CD-ROM support does *not* support LAN sharing of CD-ROM applications designed using the MSCDEX API.

It is possible to add a third-party CD-ROM LAN solution that will remedy this problem. Products are available from Online Computer Products (Opti-Net), Meridian Data (CD-Net), and CBIS (CD Connection) that resolve the MSCDEX API problem. These software add-ons generally require that the workstation/server become a dedicated server. When a workstation/server must become a dedicated CD-ROM server, a new PC must be purchased, which becomes a workstation.

Before embracing a decentralized (peer-to-peer) approach, organizations should understand that hidden costs and incompatibility issues can crop up.

Centralized Configuration

The Centralized approach suggests that all CD-ROM drives will be attached to one machine, known as a dedicated CD-ROM server (see Figure 2). Since the machine is dedicated, its only functions are to manage the multiuser access to the CD-ROM drives and reliably transfer information to the workstations on the LAN. Generally, the CD-ROM server is a wide-chassis 80386- or 80486-based PC. Given adequate free slots inside the CD-ROM server, three or more CD-ROM controllers can be installed, each with up to eight external, daisy-chained CD-ROM drives. The CD-ROM server should also be configured with a minimum of 8MB of memory, optimizing its performance for disk caching.

Disk caching is very important when one CD-ROM server must manage simultaneous, shared access to numerous CD-ROM databases. Why is disk caching important? The average access time of a CD-ROM drive is extremely slow—around 0.3 second. Without disk caching, the optical server must perform a read of the CD-ROM disc each time a user requests a record. Consequently, the user is likely to experience a delay every time the "browse" key is pressed.

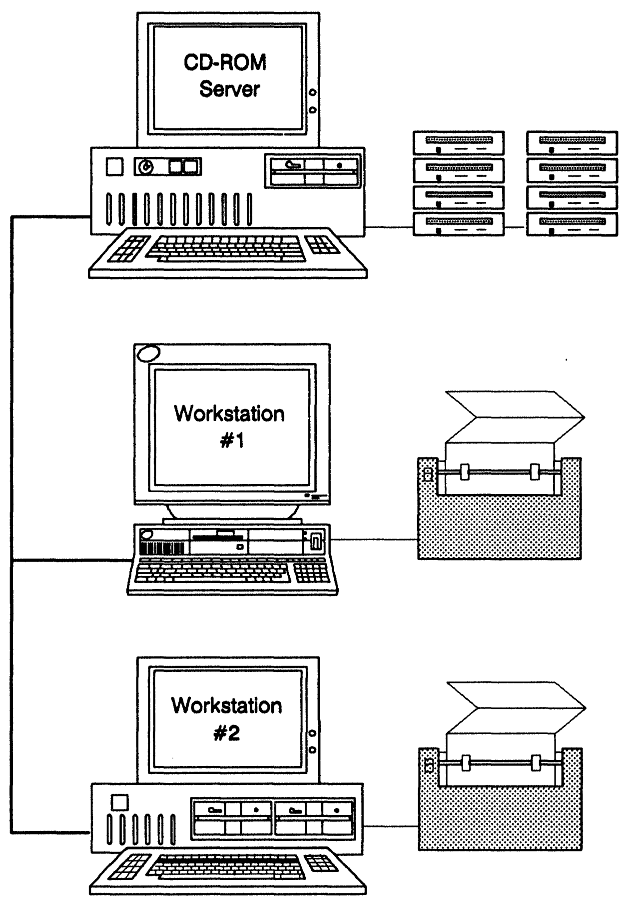
Disk caching enables the CD-ROM server, on its first read of the disk, to bring a large number of related/surrounding records into memory (RAM cache). Since related records are in memory, a browse will often find the needed record in memory, not requiring the CD-ROM server to read from disk. Multiply this small savings by hundreds of occurrences and disk caching really improves performance.

The CD-ROM server and disk drives can be moved to a secure area, behind closed doors. Workstations access the CD-ROM server over the LAN cabling. With this configuration, a single workstation malfunction or reboot will not bring the entire network down, as will happen with a Decentralized (peer-to-peer) configuration.

Limitation

The centralized approach is also susceptible to a single point of failure—the dedicated CD-ROM server. If the

Figure 2.
Centralized CD-ROM LAN Configuration



Using the centralized approach, all CD-ROM drives are attached to one machine, the CD-ROM server.

CD-ROM server goes down, all access to CD-ROM databases is denied. Planning for system redundancy will correct this weakness. One workstation should be configured with the same components as the CD-ROM server. This workstation now takes on the additional role of alternate CD-ROM server. When the CD-ROM server malfunctions, the user merely moves the controller card(s) and CD-ROM drives to the alternate server—this can be done in less than one-half hour. (System redundancy can be inexpensively accomplished by "building-your-own CD-ROM server," rather than purchasing a "tower" unit.)

Software-Only Products

As mentioned earlier, three major commercial CD-ROM networking software-only products are available: CBIS's CD Connection; Meridian Data's CD-Net, and Online Computer Products' Opti-Net. Each of these vendors provides both NETBIOS and NetWare versions. Further, these products attempt to support all CD-ROM applications, including those applications developed with MSCDEX's API.

All these products allow more than one dedicated CD-ROM server on the LAN, therefore allowing hundreds of CD-ROM drives to be placed on the LAN. CBIS and Meridian Data also provide tower units that are a turnkey CD-ROM server hardwarc/software solution (dedicated CD-ROM server with a bank of internal CD-ROM drives).

Configuring Workstations to Access the CD-ROM Server

Before a workstation can access a CD-ROM disk that has been placed on the CD-ROM server, it must be able to access and communicate with the CD-ROM server.

Workstations on the LAN requiring access to the CD-ROM server must load the following software:

- The MS-DOS operating system, including files and buffers
- The network communications protocol (IPX or NETBIOS) and a network redirector (discussed later)
- Workstation CD-ROM network software supplied by one of the commercial CD-ROM networking vendors (used to communicate with the optical server)
- MSCDEX (for network access to the applications requiring MSCDEX API support)

With all of this software loaded into the first 640KB of a workstation's conventional memory, it is not unusual for the workstation to be left with a paltry 475KB. Since many CD-ROM applications suffer from RAM cram, any workstation left with 475KB will probably only be able to run half of the CD-ROM applications on the market. Therefore, workstation memory must be optimized!

Optimizing Workstation Memory (Managing RAM Cram)

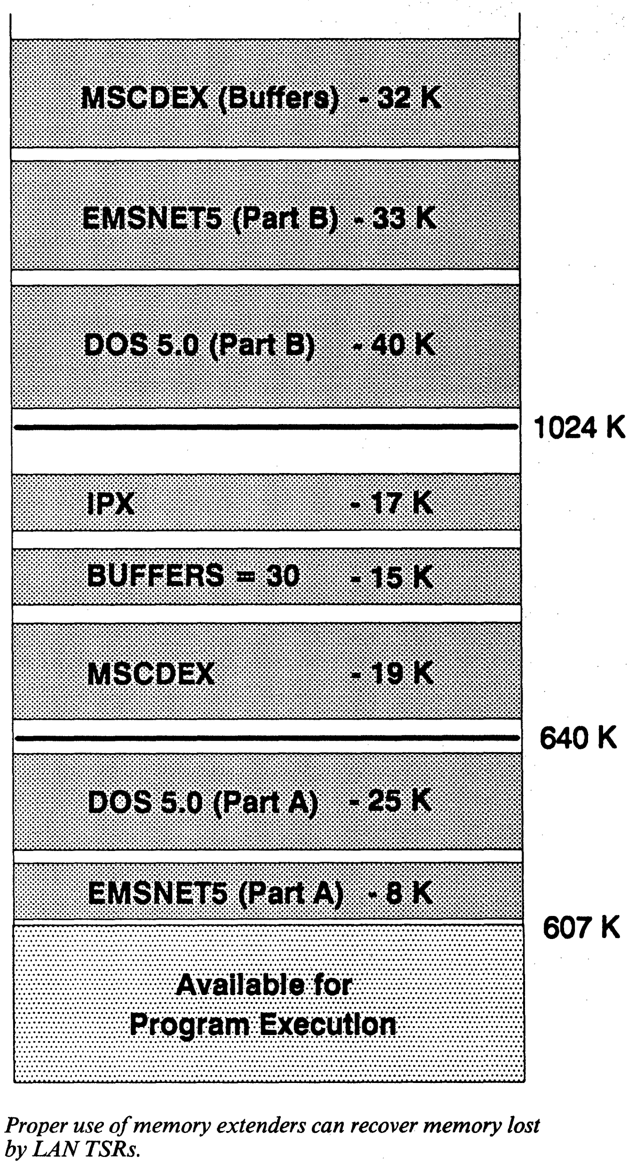
Much has been written about MS-DOS 5.0 and its ability to increase RAM through memory optimization utilities. Known as DOS Extenders, this technology is a new feature of MS-DOS 5.0, but has been available from vendors such as Quarterdeck (QEMM), Qualitas (386Max & Blue Max) and Helix (NetRoom). Proper use of memory extenders can recover memory lost by LAN TSRs. This is accomplished by loading TSRs into reserved (but exploitable) memory between 640K and 1024KB (Figure 3).

A new feature unique to MS-DOS 5.0 is its ability to load 40KB of the MS-DOS operating system between 1024K and 1088KB, increasing available workstation memory by an additional 40KB.

To utilize DOS extenders, workstations should be 386 or 386SX PCs with a minimum of 2MB of RAM. Workstations that do not have 386 processors require LIM 4.0 EEMS memory boards in addition to memory-extender software which supports 286 PCs.

It is critical to optimize workstation memory before CD-ROM applications are added to the network. The author considers 540KB of free memory (after loading all LAN TSRs) to be a minimal goal. Novell has provided leadership in this area by releasing extended/expanded memory support for its NETX.EXE redirector TSR. By using EMSNETX/XMSNETX, the NetWare redirector loads above 1024KB (more about redirector software later).

Figure 3.
Optimizing Workstation RAM



Proper use of memory extenders can recover memory lost by LAN TSRs.

MSCDEX Memory Requirements

To operate properly, MSCDEX must be configured with its own buffers. Although there are no clear guidelines, vendors often suggest 8KB of memory per disk times the largest disk set. In our previous example, the four disk set would use 32K of RAM (8KB x 4) for MSCDEX buffers. Fortunately, a recent version of MSCDEX allows the buffers to be loaded into expanded memory (above 1088KB) if available in the workstation (see Figure 3).

Configuring CD-ROM Applications for LAN Use

Before running the CD-ROM application from your workstation, the application must be directed to access the correct disk on the CD-ROM server. Since the CD-ROM drive is no longer physically connected to the workstation, a command must be made to make the proper CD-ROM drive on the CD-ROM server appear to the application as if it were directly attached to the workstation. This process of making shared network devices look like they are locally attached is known as redirection. The redirection process works as follows.

- *Step 1:* The workstation loads MSCDEX and, depending upon the parameters used, MSCDEX creates and reserves one or more drive letters for generic CD-ROM use. The letters assigned to MSCDEX begin immediately after the last hard disk letter in each workstation. If a workstation has a C: hard disk, MSCDEX letters would begin at the D: drives.
- *Step 2:* The workstation's CD-ROM network software is called, which redirects a disk on the CD-ROM server to the D: workstation letter created in Step 1.

At this point, the CD-ROM search/retrieval software (on the C: drive of the workstation's hard disk) can be configured to look for the CD-ROM disk in the D: drive. Once the application is configured, the batch file which starts the CD-ROM application should follow four steps.

- *Step 1:* Call the workstation's CD-ROM network software and map the proper disk to the proper MSCDEX drive letter.
- *Step 2:* Call/launch the CD-ROM application from the C: drive.
- *Step 3:* As the application runs, it will create spill files on the C: drive.
- *Step 4:* When the user exits the application, the batch file should include the necessary syntax to un-map the CD-ROM server's disk from the MSCDEX D: drive letter.

If the CD-ROM server has been configured with 32 CD-ROM drives, it would be logical to assume that one drive letter would be required for each CD-ROM drive to be shared. To begin, MS-DOS cannot deal with 32 letters. Fortunately, the letters that MSCDEX creates can be assigned and re-assigned, allowing the letters to be shared by many different CD-ROM applications. When setting up CD-ROM LAN access, it is important to create/reserve adequate drive letters for the largest contiguous set of CD-ROM disks that comprise a single application.

CD-ROM Applications that Span Multiple Disks

If a CD-ROM database is comprised of four disks, the database can operate as four separate applications. Users wishing to search for information which may be contained on all four disks would search each disk separately. This is both a time-consuming and cumbersome process.

Recently, some CD-ROM developers have modified their search/retrieval software so that a search can automatically span all (or a user-defineable number) of the disks in a set. Using the example of a four-disk "spannable" database, it would be necessary to create four drive letters when loading MSCDEX. If a workstation had C: as the local hard disk, MSCDEX would create the D:, E:, F:, G: drives. To properly plan for the expansion of the database, the LAN administrator would reserve additional letters.

Since the MSCDEX drive letters can be unmapped and remapped to different CD-ROM drives, CD-ROM applications with a single CD-ROM disk can be dynamically assigned to the D: drive before the application is called.

The need for CD-ROM drive letters generally comes as a real surprise for LAN administrators that have already planned out their drive letters for file server and other LAN resources. With larger disk sets, a real competition for drive letters is created. Recent upgrades to the MSCDEX software allows the CD-ROM drive letters to be started at the drive letter of the LAN administrator's choice. Any letters above E: generally require that the workstation's CONFIG.SYS file contain a LASTDRIVE statement to increase local drive letters.

When there are not enough drive letters to go around, methods must be found to interactively share the letters between CD-ROM applications and other LAN resources. Unfortunately, MSCDEX does not yet have utilities which allow it to release its hold on the drive letters that are assigned to it. The "three-finger salute" (re-boot) is sometimes required to properly reset the DOS environment.

How a File Server Fits Into CD-ROM LAN Access

Up to this point in the report, workstations running CD-ROM applications have run those application's search/retrieval software from the C: drive of their hard disks. The C: drive is also where the spill files have been created.

When a large number of workstations require access to CD-ROM applications, the idea of loading and maintaining the search/retrieval software for each CD-ROM application on everyone's hard disk becomes a nightmare for the LAN administrator. Why not install each CD-ROM application's CD-ROM search/retrieval software one time on the file server? When more than one isolated workgroup requires access to CD-ROM, there is no doubt that centralized maintenance of the CD-ROM search/retrieval software from the file server is preferred.

Some developers continue to provide an install program which will only install on the C: drive. When encountering these applications, beware that the application is likely to be "network unaware."

Applications that are network aware will provide the following user-definable parameters.

- Allow the LAN administrator to define what drive letter and/or subdirectory to which the search/retrieval software will be uploaded

- Allow the LAN administrator to define what drive letter and/or subdirectory the application will use to create spill files

There are a number of applications that require the application be started in the directory where the search/retrieval software is located. These applications often default to creating spill files in the same directory. Guess what happens when the second user tries to use the CD-ROM application and share the same directory on the file server? The CD-ROM application tries to create spill files for the second user and erases the spill files created for the first user, crashing the application for both users!

Some developers have gotten around this problem by using a DOS runtime variable (using the syntax [SET=variable—name=value]). The value of this variable can be changed to point to a common drive letter on the file server (the user's network home directory) or back to the user's C: drive.

Based on our recent work with a variety of CD-ROM applications, we estimate that 20% of the CD-ROM applications will not perform properly on a file server. Because of this, it is important to have hard drives in each workstation that will access CD-ROM applications. To support diskless workstations on a LAN, it may be essential to create a RAM disk, thereby creating a virtual C: drive. It is also possible to map a network drive to override local drive letters.

Recent third-party products allow for CD-ROM drives to be directly connected to the NetWare 3.11 file server. Using NetWare Loadable Modules (NLMs), these drives emulate NetWare (hard disk) volumes, allowing users to access the CD-ROM data from the file server.

Depending upon the CD-ROM product, it is technically possible to upload the data and index files on the CD-ROM disk to a large magnetic disk on the file server. With all of the data on the magnetic disk, the application would run substantially faster. Recognize, however, that many CD-ROM developers/publishers are reluctant to offer their information via file server access. Their reluctance is driven primarily by software piracy issues.

Multimedia

CD-ROM applications have begun to embrace multimedia, incorporating text, graphics and audio to enhance the application. Multimedia applications create integration challenges when implemented on a LAN.

- VGA and super VGA images are large files which create increased LAN cable traffic, causing potential bandwith bottlenecks.
- Audio and stereo applications (when used standalone), require the user to merely plug into the audio jack of the CD-ROM drive. LAN users, not having access to the CD-ROM drive's audio jack, must have a hardware board or software utility in their PC to capture and reproduce the audio signal.
- Full-motion video, besides creating LAN traffic, may require that PCs be equipped with DVI (Digital Video Interactive) hardware boards. These boards will compete for limited expansion slots on the PC's motherboard and will be expensive.

Summary

The intent of this report is to provide the reader with a clear understanding of the issues and obstacles associated with implementing CD-ROM applications on a LAN. The focus is on the key issues which LAN administrators and management will face when considering adding CD-ROM applications to an existing or new LAN. (UNIX, VAX, and other network configurations are not within the scope of this report.)

CD-ROM local area networking is a technology whose time has arrived. Just as Compact Disc audio caught on when enough music was published on CD, CD-ROM appears to have reached its threshold, with more than 2,000 different titles available. U.S. Government agencies have mandated that CD-ROM will supplement and/or replace microfilm, microfiche and 9-track tape. The costs associated with disseminating information on CD-ROM have continued to decline, as has the price of CD-ROM drives.

The key complaint that we continue to hear from users of CD-ROM applications is that most applications sport a different user interface. Few CD-ROM applications have been developed for Microsoft Windows. Since there are so many publishers, no agreement has been reached regarding what the F1 key or the ESC key should mean. This leaves everyone desiring a single/interchangeable search engine.

It is encouraging to know that there are organizations working to promote standards. One such organization is SIGCAT (Special Interest Group for CD-ROM Applications and Technology). "SIGCAT is a user group sponsored by the U.S. Geological Survey which is devoted to the investigation of CD-ROM technology and its myriad applications," according to SIGCAT literature. Membership in SIGCAT is free and includes bi-monthly meetings in Reston, VA and occasional meetings on the West Coast. If you cannot attend the meetings, minutes are mailed to you. Prominent speakers from around the world address the SIGCAT membership. SIGCAT has many subcommittees which focus on specific issues involving CD-ROM technology. For more information about SIGCAT, write to: E.J. (Jerry) McFaul, Chair, SIGCAT, U.S. Geological Survey, 904 National Center, Reston, VA 22092-9998.

Summary of LAN Obstacles

Restrictive LAN Licensing: Although most publishers offer a LAN license, their definition of a LAN often envisions all workstations being located inside four walls. Many publishers have failed to recognize campus and multi-building networks as LANs and do not license their products for these environments.

Cost of LAN License: Assuming that your planned use for CD-ROM products falls within the publisher's definition of a LAN, subscription fees are usually higher.

Workstation RAM Cram: Many CD-ROM applications require more than 500KB of free conventional memory. Workstation upgrades are almost always necessary.

Non-IBM PC Workstations: Don't forget that 95% of the CD-ROM applications are available only in the MS-DOS format. Macintosh, UNIX, and VT terminal users cannot access these products unless third-party add-ons are provided. These add-ons could take the form of UNIX VP/IX functionality, LAN/TCP gateways, PC/LAN node software emulation, etc.

Network "Unaware" Applications: These are applications that fail to allow the LAN administrator to upload and maintain the application's search/retrieval and spill files on the file server.

MSCDEX API Compatibility: Some CD-ROM applications require that MSCDEX (or full-emulation support) be loaded into workstation memory.

Diskless Workstations: LAN unaware applications try to write their spill files to a local C: drive. Provision must be made to provide workstations with a C: drive.

Full Multimedia Support: Increased LAN traffic, audio, and DVI support are real issues.

Increased LAN Traffic: If the LAN is already suffering from traffic-saturation, adding CD-ROM applications will only aggravate the situation. The effective use of routers is often the answer to LAN backbone saturation.

Lack of Common Interface: The most agreed upon end-user problem is the number of different user-interfaces that must be learned to use a variety of CD-ROM applications. ■

This Datapro report was written by Howard McQueen, president of CD Consultants Inc. (CDC), a consulting firm which has specialized in LAN and WAN implementation of CD-ROM applications since 1987. CDC conducts workshops on CD-ROM LAN/WAN technologies and issues throughout the country. Major consulting clients include federal government agencies, libraries, universities, hospitals and research institutions. CD Consultants is located at 4404 Keswick Road, Baltimore, MD 21210; telephone (301) 243-2755, facsimile (301) 243-9419.

Migrating LAN Workstations from DOS to OS/2

In this report:

This report will help you to:

- Migrate DOS-based IBM Token-Ring Network workstations to IBM OS/2 EE 1.1.
- Review the configuration issues involved with the migration.
- Understand the minimum hardware and software requirements.

Introduction

The hard disks of DOS workstations are typically loaded with programs and data. The invasion of more than 20MB of OS/2 EE 1.1 operating system software has a dramatic impact on the workstation. Where should this OS/2 software reside? Should it intermingle with the DOS programs and data? What resources are required to run it? What is involved in installing the OS/2 operating system software? How should you configure the OS/2 system to support LAN-based communications? How do OS/2 applications fit into the picture?

This Datapro report is a reprint of "Hewing The Trail" and "Pitching Camp" by Robert Panza, pp. 94-104 from *LAN The Local Area Network Magazine*, Volume 5, Number 3, March 1990 and pp. 87-100 from *LAN The Local Area Network Magazine*, Volume 5, Number 4, April 1990. Copyright © 1990 by Miller Freeman Publications. Reprinted with permission.

Ed Scannell, who writes for Infoworld, quite accurately described OS/2 as "the eternal beta." It constantly evolves. Therefore, any migration strategy demands flexibility. Within a very short timeframe, we have already seen OS/2 versions 1.0 and 1.1 under Standard and Extended Editions. IBM has released many Corrective Service Diskettes (CSDs) to rectify a large number and variety of bugs.

In spite of the dynamic evolution, complexity, and bugs, users who work with OS/2 fall in love with it, especially under Presentation Manager on EE 1.1 and 1.2. DOS can no longer satisfy them. Most impressive are the powerful new OS/2 applications such as PageMaker (see Figure 1), Excel, Lotus 1-2-3, Microsoft Word, and SQL front-end applications. Presentation Manager and the communication features of the OS/2 workstation are irresistible lures. Migration from DOS to OS/2 will soon become commonplace for most upscale business users.

The tactical provisions of migrating LAN workstations from DOS to OS/2 are fourfold: 1) take full advantage of OS/2 capabilities, 2) organize DOS workstation applications not available for OS/2 to run in the DOS Compatibility Box, if possible, 3) provide a DOS boot fall-back provision as a contingency for workstation applications not running under OS/2 or in the DOS Compatibility Box, and 4) find a way to back up the OS/2 workstation.

Resource Requirements

According to the IBM pamphlet in the OS/2 EE 1.1 documentation set, "Installing OS/2 Extended Edition," the following *minimum* hardware and software are needed to support OS/2 EE 1.1:

- An IBM PS/2 or PC with an 80286 processor, such as Models 30-286, 50, 50Z, 55, 60, 70, or 80; an IBM PC-AT; or an IBM PC-XT Model 286.
- 3MB of RAM.
- 20MB of fixed disk storage space.
- A LAN adapter, if the computer will be a LAN workstation.
- One or more communications adapters, if using the Communications Manager.
- A modem, for remote communications.
- A high capacity diskette drive, either 5¼ or 3½ inches.
- A keyboard, display, and display adapter.

In reality, these resources are a bit skimpy. A fast PS/2 should have at least 4MB of RAM. The PC-AT should have a minimum of 5MB to 6MB of RAM to compensate for its slower processor, RAM, and disk drive. When OS/2 starts to swap-out from RAM to disk, the PC-AT workstation comes to a halt. Swapping out is not a problem for fast PS/2s. A 6 MHz PC-AT with a 20MB hard disk is a bit too slow and limited in disk storage for OS/2 EE 1.1 in my judgement. But you can adapt the 8 MHz PC-AT with a 30MB hard disk to OS/2 EE 1.1 surprisingly well.

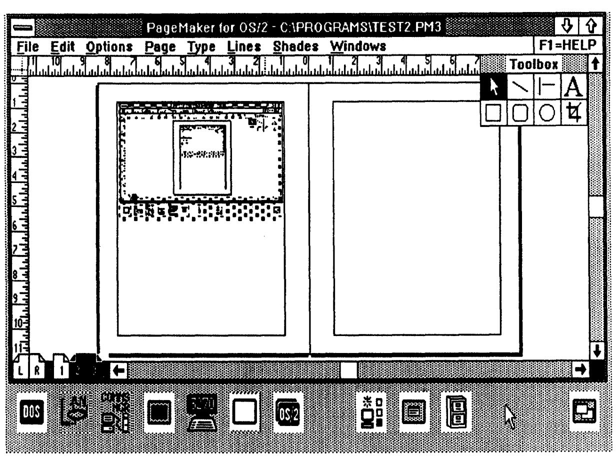
A hard disk with only 20MB of disk space is also inadequate since the OS/2 EE 1.1 software will practically take up all of the disk. In order to run OS/2 EE 1.1 on an 8 MHz PC-AT with a 30MB hard drive, I had to relocate most of my local application programs and data to a file server. While my workstation philosophy advocates using server storage rather than local workstation disks as much as possible, I think that a 60MB local hard drive is appropriate for an OS/2 EE 1.1 workstation.

Therefore, any 80286 OS/2 EE 1.1 workstation should have at least a 10 or 12 MHz processor and minimum local disk storage of 40MB to 60MB. While the faster 80286-based computers have sufficient processor horsepower for running OS/2, 80386-based computers should be purchased instead, if at all possible, given that the 80386SX models are so reasonably priced.

My OS/2-LAN enthusiast friends on Wall Street prefer fast 80386- or 80486-based workstations with local hard drives in excess of 100MB. They know that the 80386 versions of OS/2 (v. 2.0) will require 80386 processors, so the future migration to the most powerful and feature rich versions of OS/2 will diverge from the 80286. The 100MB-plus local disk offers a lot of utilitarian value to aggressive workstation users.

Don't forget the mouse. The PM interface without a mouse is a nuisance to use because of the

Figure 1.
OS/2 Application with Muscles



PageMaker for OS/2 is one of the most powerful OS/2 applications—more powerful than on the high-end Macintosh. The PM interface also has advantages over that of the "big Macs." You can edit/process several documents at the same time with ease, while other applications simultaneously do their own work. Postscript printer support over the LAN is standard on the OS/2 LAN workstation.

cumbersome and illogical keystrokes required. All OS/2 workstations running PM must have one to reap the operating efficiency of the PM interface.

Migration Overview

In migrating our hypothetical LAN workstation from DOS to OS/2 EE 1.1, we will go sequentially through the following steps in detail:

- Relocate all of the programs and data in the DOS partition on the workstation C: drive, if that partition runs under DOS 3.30 (or below).
- Build a DOS boot floppy as a fallback for running DOS applications that are important to the workstation user.
- Install the OS/2 Base Operating System software.
- Install the OS/2 Extension software modules, i.e., Communications Manager, LAN Requester, and Data Base Manager.
- Apply the latest IBM CSD for OS/2 EE 1.1.

Relocating DOS

OS/2 EE 1.1 installs to the C: partition, where most of the user's DOS programs and data probably now reside. The most likely problem here will be insufficient space in the partition for both the DOS and OS/2 files. Users generally load up the C: drive with programs and data before expanding to other workstation drives.

DOS 3.30 users normally have their disk divided into 32MB partitions. In the migration of a DOS 3.30 workstation from DOS to OS/2, I usually XCOPY the DOS programs and data from the C: to the D: drive. That gives OS/2 an entire 32MB partition to take over. If the workstation does not have enough local disk storage to support the XCOPY maneuver, then you must move the files to shared storage devices on the LAN.

Users running DOS 4.01 have more options open. Since the size of a workstation disk partition is no longer limited to 32MB, the partition can hold both OS/2 and DOS file sets. A PS/2 with a 115MB hard disk may be configured with the full 115MB in a single partition. This doesn't solve the dilemma of co-mingling DOS and OS/2 files, however.

Beginning with OS/2 v. 1.2, workstation partitions will use either 1) the file allocation table (FAT) file system of DOS or 2) a new High Performance File System (HPFS) which can not be locally accessed under DOS. In other words, partitions using the new OS/2 non-FAT file systems, such as the HPFS of v. 1.2, will likely be inaccessible under DOS. This might change in future releases of DOS, such as DOS 4.10, but we have no way of knowing for sure until that happens.

In formulating our migration strategy, we must never forget that OS/2 will continue to evolve very rapidly. Flexibility in the workstation is very important. New installable file systems (IFS) in the OS/2 environment may or may not be fully compatible with DOS or with each other.

For now, I have been dedicating a partition to DOS if possible, and likewise I anticipate having multiple partitions on the workstation in the future to accommodate different file systems, such as HPFS-286 and HPFS-386. Separate partitions are also very helpful for isolated testing of newly released file systems.

With IBM OS/2 EE 1.1, users must boot DOS off a floppy disk. The Microsoft OS/2 dual boot option was removed by IBM. A dual boot option gives the user a choice of booting from either DOS or OS/2 upon initialization of the workstation. Once OS/2 EE 1.1 is installed, the workstation hard disk will only boot OS/2. IBM's OS/2, v. 1.2 will supposedly bring back the dual boot option, effectively making it unnecessary to use a DOS boot floppy.

Since the DOS-based workstation was XCOPYed from C: to D:, you must consider the impact of the change in drives. An elaborate DOS-based workstation usually has the drive designation hard coded in numerous BAT files, menu- or windows-based workstation commands, scripts, environmental variables (such as PATH and COMSPEC), etc. After booting off the DOS floppy, use the following commands to adjust the DOS workstation software to the D: drive.

```
SET COMSPEC=D:\DOS\COMMAND.COM
ASSIGN C=D
```

COMSPEC tells the operating system to reload the command processor from the D: drive rather than from the boot floppy in A:. ASSIGN instructs DOS to route all disk I/O requests for C: to D:. All BAT files and other workstation commands with hard

Figure 2.
From Hypothetical OS/2 EE 1.1 Workstation

CONFIG.SYS

```
BUFFERS=16
FILES=30
SHELL=D:\DOS\COMMAND.COM  /E:384 /P
DEVICE=D:\LANSUPP\DXMAOMOD.SYS  001
DEVICE=D:\LANSUPP\DXMCOMOD.SYS  40000000107
DEVICE=D:\LANSUPP\DXMTOMOD.SYS ST=2 S=3 C=2 ES=2 EST=2 TC=3 0=N
BREAK=ON
```

AUTOEXEC.BAT

```
@ECHO  OFF
SET  COMSPEC=D:\DOS\COMMAND.COM
CALL  D:\BAT\LANPATH.BAT
YNPROMPT  Y  N  32  Start  PC  LAN  Program  1.30  (Y/N)?
ECHO.
IF  ERRORLEVEL  1  GOTO  NOPCLP
NET START  RDR  WORKST1 /SRV:2 /SES:3 /CMD:2 /NBC:2 /NBS:512 /PB1:80 /PB2:80 /PB3:80
:NOPCLP
ASSIGN  C=D
:EXIT
```

coded C: do not have to be changed. The DOS workstation will run flawlessly from the D: drive.

Figure 2 shows examples of the AUTOEXEC.BAT and CONFIG.SYS from the DOS boot floppy of a hypothetical OS/2 EE 1.1 workstation. Notice that the SHELL in the CONFIG.SYS file points to the same command processor as COMSPEC. Remember that we booted off a floppy in the A: drive, so COMSPEC = A:\COMMAND.COM until we change it in the AUTOEXEC.BAT file's second line. Also, notice that the execution of ASSIGN C = D comes after initializing the LAN workstation software via the NET START line. The DOS workstation in the example happens to use the IBM PC LAN operating system.

In the migration from DOS to OS/2, technical support personnel or the user must first ensure that the workstation functions fully after relocation of the DOS files formerly stored in the C: partition. Thoroughly test the workstation, booting off the DOS floppy and running from the D: partition, before installing additional OS/2 software.

OS/2 Base Operating System Installation

Installing the OS/2 EE 1.1 software is fairly straightforward. The base operating system alone needs about 8,510,000 bytes of disk space for its 207 user files, three hidden files, and six directories.

After copying the DOS files from the C: partition to the D: partition and testing all DOS operations, boot off the OS/2 installation diskette in the A: drive and follow the instructions displayed. I always select the option to reformat the C: partition. The reformatting removes the DOS files from C: as well as initializing and preparing the partition for the OS/2 software. Some OS/2 files are copied from the installation diskette. The first part of the Base Operating System installation ends when the system requests the installer to remove the installation diskette from drive A: and reboot.

Next, the OS/2 system initializes from the C: drive for the first time, and asks the installer to insert diskettes into drive A:, one by one. The system requests diskettes one through four if using 5¼-inch floppies. Files are copied to the C: partition from the floppies. An option screen allows installation of device drivers from a Device Support Diskette; an Install Device Drivers menu facilitates automatic installation of non-IBM device drivers or IBM device drivers not included with the OS/2 EE 1.1 package. Screens also appear for selecting country and keyboard, mouse, and serial device support.

The system will be configured using the most commonly installed features and configuration settings as defaults. The installer may review default settings to accept or change them. After the installer accepts the configuration settings, the system completes installation of the base operating system. The installer then removes the last floppy and reboots the workstation. OS/2 initializes off the C: drive and brings up the Start Programs window, the DOS Compatibility Box, the Print Spooler, and the Task Manager.

Start Programs shows the Main Group, which is a kind of Presentation Manager main menu or main window (see Figure 3). At inception, the

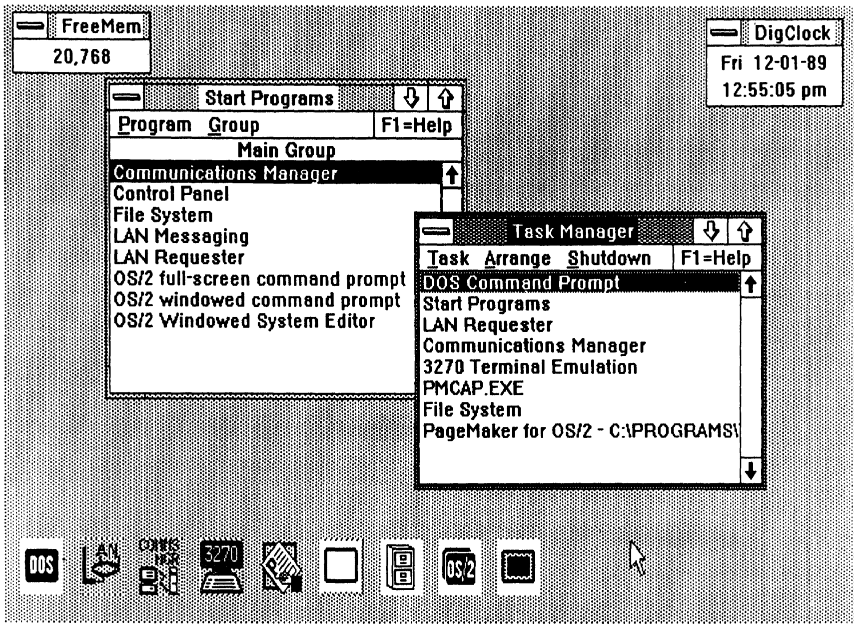


Figure 3.
Concurrent Control

The Presentation Manager interface gives the workstation user command of all concurrent processes. Each process is an active program. Across the bottom of the screen from left to right, the icons represent the DOS Compatibility Box, the IBM LAN Requester, the Comms Manager, a 3270 terminal emulation session, PageMaker, a Presentation Manager screen capture utility, an OS/2 File System utility, an OS/2 full screen prompt session, and an OS/2 window prompt session. A double click on any of the icons brings the process represented by the icon into the foreground. The OS/2 Extension module icons were created by Kerry Gerontianos, a leading edge OS/2 application designer in New York City.

main group of Start Programs offers four selections: a File System utility, the Introducing OS/2 online tutorial, OS/2 full-screen command prompt, and OS/2 windowed command prompt. A second group, called Utility Programs, includes Control Panel, Disk Information—CHKDSK, Format Diskette, and OS/2 System Editor.

We now have a working OS/2 workstation, although there will not be any LAN, communication, or database functionalities until after the installation of the proper Extension modules.

Installation of the OS/2 Extension Modules

The next steps are to run individual installation jobs for the OS/2 Extension modules: Communications Manager, LAN Requester, and Data Base Manager.

The Extension modules conform to IBM's full screen System Application Architecture (SAA) interface and support the mouse pointing device. However, they are not yet PM applications that will run in windows.

In addition, I always install Borland's SideKick, which comes packaged within the OS/2 EE 1.1 shrink-wrap as though it were meant to be one of the Extensions. SideKick for OS/2 EE 1.1 is an example of a real PM application.

Comms Manager

All IBM OS/2 EE 1.1 communications functionality stems from the Communications (Comms) Manager. The Comms Manager software installs into the C:\CMLIB subdirectory. After double clicking on the OS/2 full screen command prompt, entering CMINST at the OS/2 prompt [C:\] starts the installation process. Reinstallation of the Comms Manager at a later time requires CMINSTM.

The installer may choose between two different methods of installation: Prompted Mode or Full Function Mode. Novices should use Prompted Mode which requires a minimum number of decisions. In reality, most users have difficulty installing and configuring OS/2 EE 1.1, and usually require expert assistance—configuration of the Comms Manager is unquestionably the most complex part of the OS/2 EE 1.1 installation process.

The menu-driven Prompted Mode installs the Comms Manager base code, copies the user configuration files or the IBM-supplied configuration files, and installs the features that are specified in those configuration files. The installer always has the option of modifying the Comms Manager configuration file after completing installation.

Full Function Mode is also menu-driven. It requires the installer to select different communication functions, specify the details of their configuration, and indicate the order of execution for functions. The installer must be prepared to provide a lot of detailed technical information.

I generally recommend Prompted Mode rather than Full Function Mode, since it is so easy to get into trouble configuring the Comms Manager. However, the configuration resulting from the Prompted Mode installation will need to be fine-tuned later.

After selecting Prompted Mode off the install menu, the system prompts the installer for OS/2 EE 1.1 diskettes in order to copy the Comms Manager software to the C: drive. With 5¼-inch floppies, diskettes five and six are requested. The menu selection, "Copy the user configuration files," provides an opportunity to input customized Comms Manager configuration files. The system administrator or technical support specialist should supply customized files, tailored for a particular group of users in advance.

Without custom configuration files the Comms Manager selects "Copy the IBM configuration files." After copying the IBM configuration files to the C: drive, the menu selection "Complete the installation" finishes the job. The history file C:\CMLIB\ACSINSUM.DAT provides a summary of current installation data, including time and date installed, file names, and directory pathnames.

The program title "Communications Manager" now appears in the Main Group list of the Start programs window. However, the Comms Manager will not function without adding the proper device driver statements to the CONFIG-.SYS file and then rebooting the workstation.

LAN Requester

The LAN Requester allows the workstation to perform LAN functions on an IBM Token-Ring or IBM PC Network. Through it you can access LAN resources, such as disks, printers, and serial devices. In addition, it can define, control, and manage access to LAN resources.

The LAN Requester files install into the subdirectory tree off \IBMLAN. It is the only OS/2 EE 1.1 module which allows installation on a drive other than C:. OS/2 EE 1.2 provides more flexibility for installation of the OS/2 Extension modules.

To begin, enter REQINST at the OS/2 full screen command prompt. Do not run any other programs in the DOS Compatibility Box or another IBM OS/2 session during installation. When asked, specify the drive and indicate whether or not to install the Online Reference and Help panels on the local drive. I usually install them.

With 5¼-inch floppies, the installation job requests OS/2 diskettes nine, 10, and 11. The system copies the files and asks for the LAN Requester name, the Domain name, and then the requester services to initialize when the LAN Requester starts. I usually specify all of the available services: Messenger and Message Pop-up. Last, select the type of network adapter: PC Network (all), Token Ring Network 16/4, Token Ring Network \A or PC Adapter II, and Token Ring Network Adapter I.

LAN Messaging and LAN Requester now appear in the Main Group list of the Start Programs window. You can display or print a history file of the installation.

Now you have to reinitialize the workstation to activate the appropriate device drivers before starting the LAN Requester. During the installation process, device drivers for LAN communications (LANDD.SYS, TRNETDD.SYS and NETBDD.SYS for the Token Ring Network) were automatically installed in the CONFIG.SYS file. They specify the default configuration file for the LAN Requester, C:\IBMLAN\REQBASE.CFG. This configuration file works with the LAN Requester but not the Comms Manager, which requires a different configuration file.

Later you must edit the CONFIG.SYS file to replace the \path\REQBASE.CFG with the proper Comms Manager configuration path and file name. Normally, the Comms Manager configuration file supports both the Comms Manager and the LAN Requester.

The installation process included other CONFIG.SYS entries, as well: 1) a device driver providing an interface for the LAN Requester (NETWKSTA.SYS) that specifies an initialization

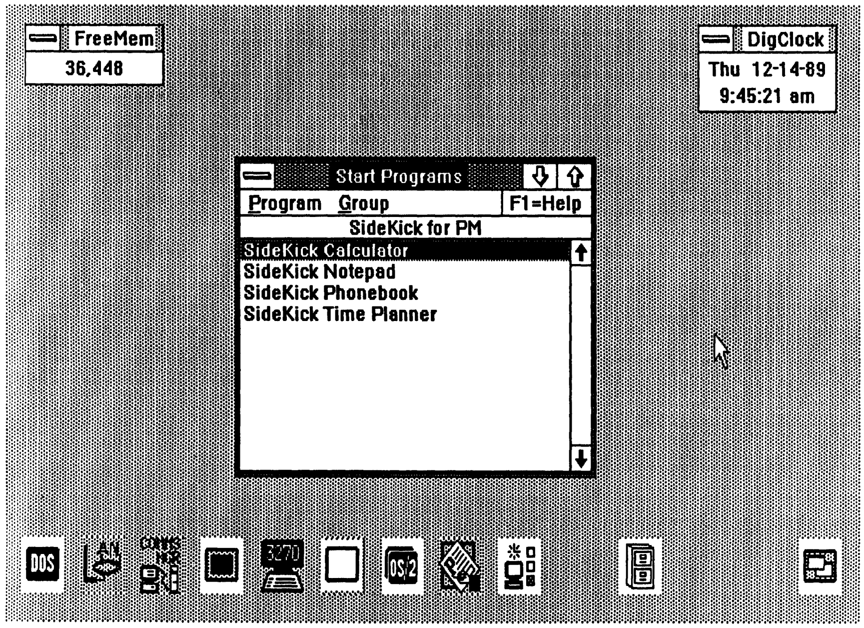


Figure 4.
SideKick

File System and Directory Tree were put down with the other icons by single clicking on the down arrow in the upper right hand corner of the File System window. The icon for the File System is the filing cabinet. Afterward, the SideKick for PM group was selected in the Start Programs window. SideKick installs into PM as a separate group with selections for Calculator, Notepad, Phonebook, and Time Planner. It is a good example of a PM application.

parameter file called IBMLAN.INI, and 2) a statement allowing use of the Token Ring system process through the adapter handler, RUN C:CMLIB\ACSTRSYS.EXE. The Token Ring system process loads when the workstation boots if it finds this statement in the CONFIG.SYS file. In OS/2, you can use the RUN command to start executable system programs from the CONFIG.SYS file.

With the Base Operating System, Comms Manager and LAN Requester installed, approximately 19,270,000 bytes of OS/2 files and 12 directories reside in the 32MB C: partition.

Database Manager

The Database Manager consists of two components: Database Services and Query Manager. Database Services is described as the engine of the Database Manager—the implementation of the relational data model as well as the base functions and configuration files needed to use the Database Manager. Query Manager provides panels and menus to guide the user through database operations, i.e., creating databases, editing data, and generating reports. In addition, Query Manager configures Database Services.

Install Database Services to drive C: in the directory C:\SQLLIB before the Query Manager. Enter DSINST at the OS/2 full screen command prompt [C:\]. As before, the program requests OS/2 floppies by number. With 5¼-inch floppies, DSINST asks for diskette seven. Copying these files completes the installation.

Afterward, edit the CONFIG.SYS file to set the environmental variable SQLUSER to a unique identifier of eight characters or less. Database Manager will assign the default, NUL-LID, otherwise.

Query Manager is optional but I recommend installing it. At the OS/2 full screen command prompt, enter QMINST to begin the installation job. Using 5¼-inch floppies, OS/2 requests diskettes eight and nine to copy files to C:\SQLLIB. The program title "Query Manager" will appear in the Main Group list of the Start Programs window now. The Database Services installation alone did not place an entry in the Main Group list. Again, a history file may be viewed and/or printed.

The disk now has all OS/2 EE 1.1 modules: Approximately 24,720,000 bytes in 654 user files, three hidden files, and 13 directories. No applications or utility programs have been locally installed yet.

SideKick

SideKick provides a good example of how a software product can install into the PM interface (see Figure 4). Start with [A:\]SETUP A: C:. Parm %1 is the floppy drive from where the programs will be installed. Parm %2 following SETUP is the drive on which SideKick will be installed, and may be set to drives other than C:. After the installation begins, two informational screens appear. Next, the system copies SideKick files to the hard disk and installs SideKick into the PM interface as the third group of Start Programs. The 5¼-inch version of SideKick has two floppies. When the installation is complete, the program instructs the installer to remove the last floppy and reboot the workstation.

Now approximately 26,900,000 bytes of files fill the C: partition, consisting of 688 user files, three hidden files and 19 directories. The software installation off floppies from Base Services through SideKick takes a bit less than two hours.

Applying the Latest CSD

The installer should apply a CSD release after the OS/2 software has been installed. IBM will continue to issue a lot of OS/2 CSDs. Each CSD is an accumulation of fixes, including those from prior CSDs. Beware that some of the IBM CSDs have introduced new "bugs." While the latest CSD at this time is CSD 3072, I have only applied CSDs through 3042.

On 5¼-inch floppies, CSD 3042 consists of an installation diskette plus seven floppies and takes about 30 minutes to install. The installer boots the workstation off the CSD installation floppy. You can indicate which disk drives the CSD should not update, and whether or not to bypass any subdirectories on the drives to be updated. The CSD must run without interruption. The installer merely changes CSD floppies, as instructed by the installation program, until finished. The file C:\FIXLOG.OS2 contains a log of the latest CSD update.

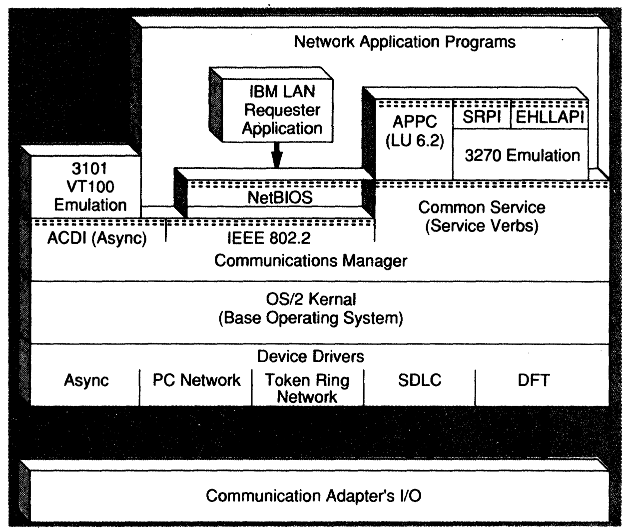
Warning: when applying a CSD after the initial installation, IBM default CFG files for the Comms Manager will be overlayed. Configuring the Comms Manager can be a pain, so backup or rename any CFG files in use that have been customized.

After the application of CSD 3042, total storage in use on drive C: exceeded 27,000,000 bytes!

Configuration Issues

Configuring the OS/2 workstation involves: 1) editing the CONFIG.SYS file; 2) tailoring the Comms Manager configuration file; 3) reviewing (and possibly editing) the IBMLAN.INI file and setting up an OS/2 LAN Requester ID, if one does not exist for the workstation user; 4) creating the STARTUP.CMD and related initialization CMD files;

Figure 5.
Network Application Programs



This diagram shows the relationship between application programs, adapter support software, and different adapters utilizing the OS/2 EE 1.1 Communications Manager interfaces. The double dashes denote the Application Programming Interfaces (APIs). Communications Manager provides comprehensive communications capabilities for a wide variety of interconnections. Since OS/2 is multitasking, the various communication options can usually run concurrently.

and 5) setting up the AUTOEXEC.BAT, other BAT files, and a menuing system for DOS Compatibility Box operations (see Figure 5).

OS/2 CONFIG.SYS File

The CONFIG.SYS file of the OS/2 LAN workstation (see Figure 6) is much more complex than that of the DOS LAN workstation. To a large degree, the differences between the two CONFIG.SYS files reflect the relaplexity of OS/2 and DOS.

In the CONFIG.SYS file, PROTSHELL loads the user interface program (Presentation Manager) and the OS/2 command processor. SET PATH establishes the search path for commands. SET DPATH gives the search path for data files outside the current directory. SET COMSPEC defines the path that OS/2 will use to reload the OS/2 command processor CMD.EXE, when necessary. SET PROMPT sets the prompt.

SET SQLUSER identifies a user of Data Base Services and Query Manager. TIMESLICE determines the minimum and maximum time value allocations of processor time to processes and programs. BUFFERS sets the number of disk buffers that the operating system uses. DISKCACHE

Figure 6.
The CONFIG.SYS File

```
PROTSHELL=C:\OS2\PMSHELL.EXE C:\OS2\OS2.INI C:\OS2\CMD.EXE
SET PATH=C:\CMLIB;C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;C:\;
      C:\IBMLAN\NETPROG;C:\CMD;C:\PROGRAMS
SET DPATH=C:\CMLIB;C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;
      C:\;C:\IBMLAN\NETPROG;
SET COMSPEC=C:\OS2\CMD.EXE
SET PROMPT=$I[$P]
SET SQLUSER=RPANZ
TIMESLICE=45, 125
BUFFERS=60
DISKCACHE=64
MAXWAIT=3
MEMMAN=SWAP, MOVE
PROTECTONLY=NO
SWAPPATH=C:\OS2\SYSTEM 512
THREADS=255
TRACE=OFF
SHELL=C:\OS2\COMMAND.COM /P
BREAK=OFF
FCBS=16,8
RMSIZE=640
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
CODEPAGE=437,850
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSEAO2.SYS
DEVICE=C:\OS2\PMDD.SYS
DEVINFO=SCR,EGA,C:\OS2\VIOTBL.DCP
DEVICE=C:\OS2\EGA.SYS
DEVICE=C:\OS2\COMO1.SYS

rem The following device drivers are for access to the
rem Token-Ring Network and are inserted by the IBM LAN
rem Requester installation program. They were modified for
rem the Comms Manager configuration file (CFG=) by the
rem installer of this workstation.
DEVICE=C:\CMLIB\LANDD.SYS
DEVICE=C:\CMLIB\TRNETDD.SYS CFG=C:\CMLIB\BOBP.CFG
DEVICE=C:\CMLIB\NETBDD.SYS CFG=C:\CMLIB\BOBP.CFG
DEVICE=C:\IBMLAN\NETPROG\NETWKSTA.SYS /I:C:\IBMLAN

rem The following line loads the Token-Ring Network process.
RUN=C:\CMLIB\ACSTRSYS.EXE

DEVICE=C:\OS2\ANSI.SYS
IOPL=YES
LIBPATH=C:\CMLIB;C:\OS2\INSTALL;C:\OS2\DLL;C:\DLL;C:\;C:\IBMLAN
      \NETLIB;C:\SIDEKICK;
```

Listing 1.—*The CONFIG.SYS File. Indented lines continue from the line above.*

(Indented lines continue from the line above.)

defines the number of storage blocks to allocate for control information and for use in disk caching. MAXWAIT determines the time limit for which a process could have lack of access to the processor resource.

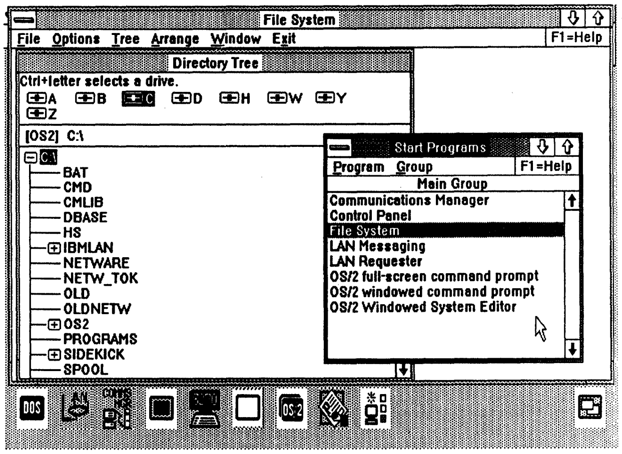
MEMMAN indicates whether there will be segment swapping out to disk and storage compaction. PROTECT-ONLY indicates whether or not the workstation will run the DOS Compatibility Box as well as OS/2 operating mode. SWAPPATH specifies the size and location of the swap file. THREADS sets the number of concurrent independent "actions" for OS/2 mode. TRACE, when set, allows the tracing of system events. SHELL loads and starts the DOS command processor, COMMAND.COM.

BREAK permits or does not permit DOS to check for CTRL + BREAK when a program requests the base operating system to perform any functions. FCBS defines the file control block information for DOS mode only. RMSIZE determines the highest storage address, in multiples of 1,024 bytes (up to 640), allowed for the DOS Compatibility Box.

COUNTRY identifies the country, for country-dependent information, and is required for code page switching. DEVINFO prepares a device for system code page switching. CODEPAGE selects the system code pages for the base operating system to prepare for code page switching.

Many device drivers are installed: POINTDD.SYS provides mouse draw support; MOUSEA02.SYS, a special Microsoft LAN Manager device driver, implements improved support for the Microsoft mouse on a PC-AT under IBM OS/2 EE 1.1; PMDD.SYS provides pointer draw device support for the Presentation Manager (PM); EGA.SYS supports the EGA register interface; COM01.SYS allows OS/2 programs to use serial ports; LANDD.SYS enables OS/2 application programs to use the NetBIOS or Device Driver interface (DDI); TRNETDD.SYS allows the workstation to access the IBM Token-Ring Network; NETBDD.SYS enables OS/2 application programs to use the NetBIOS application programming interface (API); NETWKSTA.SYS provides an interface for the IBM LAN Requester; and

Figure 7.
OS/2 File System



The OS/2 File System utility is used to view the various disk drives. From the Directory Tree you can copy to files, directories, or a tree of directories. A double quick click on the Start Programs icon brings its window to the foreground, as shown here.

RUN=path\ACSTRSYS.EXE loads the IBM Token-Ring process whenever the workstation is initialized.

ANSI.SYS supports extended display and keyboard support mode. IOPL allows I/O privilege to be granted to requesting processes in OS/2 mode. LIBPATH identifies the locations of dynamic link libraries for OS/2 programs.

Notice that a CFG= at the end of the TR-NETDD.SYS and NETBDD.SYS device driver lines defines the Comms Manager configuration file. Those lines of the CONFIG.SYS file required editing to change the CFG= from C:\IBMLAN REQBASE.CFG to C:\CMLIB\BOBP.CFG. The LAN Requester installation process writes out C:\IBMLAN\REQBASE.CFG by default.

The Base System Services installation had created a basic CONFIG.SYS file, which excluded only the five lines beginning with DEVICE=C:\CMLIB\LANDD.SYS and ending with RUN=C:\CMLIB\ACSTRSYS.EXE. When Comms Manager, Database Manager and SideKick were installed, their respective installation programs updated CONFIG.SYS only by adding their directories to the SET PATH=, SET DPATH=, and LIBPATH= lines. The LAN Requester also updated those path lines, and in addition, wrote out the LAN-related device drivers to the CONFIG.SYS file.

At the end of the NETWKSTA.SYS line, a \l: flag indicates the location of the IBMLAN.INI file. IBMLAN.INI contains all of the initialization parms for the LAN Requester.

Under OS/2, notice that remarks inserted into the CONFIG.SYS file won't cause an error message like DOS gives.

Comms Manager

After properly defining the CONFIG.SYS, edit the Comms Manager configuration file (BOBP.CFG in the example or ACSCFGUS.CFG by default). The Comms Manager CFG file will be unique in part for each workstation regardless of the similarities between them.

The IBM OS/2 EE 1.1 User Guide provides worksheets for defining the configuration values. To reiterate, configuring the Comms Manager is the most complex part of the OS/2 EE 1-1 installation, although IBM will make it less complex under OS/2 EE 1.2.

From the Communications Main Menu, select "Advanced" and then "Configuration" to arrive at the Communication Configuration Menu. Next you have to choose a communication configuration file name. If the system administrator has not provided a CFG file for the Comms Manager installation, start with the default file ACSCFGUS.CFG (US keyboard). After completing and testing the configuration, rename ACSCFGUS.CFG to a unique file name, such as BOBP.CFG.

In order to define a LAN workstation that runs the LAN Requester and 3270 terminal emulation over the Token-Ring Network, define the following profiles:

Workstation Profile

This profile includes machine type-model number, serial number, translation table file name, error log file name, error log size, error log overflow option, message log file name, message log size, message overflow option, and whether or not to auto-start features. Select auto-start for the following features: ACDI service, ASCII terminal emulation, DFT 3270 terminal emulation, and 3270 terminal emulation for IBM Token-Ring Network/SDLC and/or APPC service. Features selected to auto-start will start automatically after initialization of the Comms Manager.

SDLC/IBM Token-Ring Network 3270 Emulation Profile

Adapter number, destination address, and Logical Unit (LU) information must be specified for 3270 terminal emulation over the IBM Token-Ring Network.

SNA Base Profile

Physical unit (PU) name, network name, node ID in hex, and whether or not to activate the APPC attach manager must be defined. PU name and node ID are unique for each workstation.

IBM Token-Ring Network DLC Adapter Profile

For Data Link Control (DLC), specify the following: adapter number, whether or not to load DLC when the Comms Manager is started, maximum number of link stations, percent of incoming calls, whether or not to free an unused link, congestion tolerance percentage, maximum RU size (the amount of data that can be transferred at one time), send window count, receive window count, C&SM (Communications and System Management) LAN ID, and whether or not to send an alert for beaconing.

IEEE 802.2 Token-Ring Profile

This profile includes: adapter number and version, adapter shared RAM address, whether or not to use the universally administered address or a Locally Administered Address (LAA), the adapter address, maximum number of Service Access Points (SAPs), maximum link stations, maximum number of group SAPs, maximum members per group SAP, maximum number of users, transmit buffer size, number of transmit buffers, receive buffer size, minimum receive buffers, adapter "open" options (whether or not to use the wrap interface, contender, and/or to override the token release default), group 1 response timer (T1), group 1 acknowledgment timer (T2), group 1 inactivity timer (Ti), group 2 response timer (T1), group 2 acknowledgment timer (T2), group 2 inactivity timer (Ti), number of queue elements, and the number of global descriptor table selectors. The installer must make sure that there is a unique Token-Ring adapter address for each workstation if using an LAA.

Figure 8.
The LAN Requester Initialization File

```
[networks]
; This information is read by the redirector at device
;initialization time.
; It is available to application programs via NetBiosEnum.

      net1=netbios$, 0, NB30, 32, 32, 16

[requester]
      computername=BOBP
      domain=DOMAIN1
; The following parameters generally do not need to be
; changed by the user.

      charcount=16
      chartime=250
      charwait=3600
      keepconn=600
      keepsearch=600
      maxcmds=16
      maxerrorlog=100
      maxthreads=10
      maxwrkcache=64
      numalerts=12
      numcharbuf=10
      numservices=8
      numworkbuf=15
      printbuftime=90
      sesstimeout=45
      sizcharbuf=512
      sizerror=1024
      sizworkbuf=4096

; The next lines help you to locate bits in the
;wrkheuristics entry.
;                       1         2         3
;              0123456789012345678901234567890l
wrkheuristics= 111111111111111111000101111201110

      wrknets=net1
      wrkservices=MESSENGER,NETPOPUP

[messenger]
      logfile=messages.log
      sizmessbuf=4096
[services]
; Correlates name of service to pathname of service program.
; The pathname must be either
;          1) an absolute path (including the drive specification)
;                        OR
;          2) a path relative to the IBMLAN root

      requester=services\wksta.exe
      messenger=services\msrvinit.exe
      netpopup=services\netpopup.exe
```

Listing 2.—*The LAN Requester initialization file.*

NetBIOS Profile

The following are defined in the NetBIOS profile: adapter number, whether or not NetBIOS is installed, whether or not to use full buffer datagrams, whether or not datagrams use a remote directory, whether or not the universally administered address is reversed, maximum link stations, maximum sessions, maximum commands, maximum names, query timeout value, transmit query count, maximum transmits outstanding, maximum receives outstanding, retry count for all stations, ring access priority for messages, number of remote names, response timer intervals, acknowledgment timer intervals, and inactivity timer intervals.

The features defined above support the LAN Requester and 3270 terminal emulation over the Token-Ring Network. Many other Comms Manager features remain undefined, such as: asynchronous features, XMODEM file transfer, TSO file transfer, DFT 3270 terminal emulation, SDLC 3270 terminal emulation, 3270 file transfer profiles, 3270 keyboard profiles and alarms, APPC features, the Server-Requester Programming interface (SRPI) profile, and non-SNA connection (through a non-SNA controller) features.

After setting up or changing the Comms Manager configuration, run the verification program to check for validity of the settings and for consistency across the various Comms Manager profiles.

LAN Requester

The IBMLAN.INI ASCII file defines the LAN Requester initialization parameters (see Figure 8). I used suggestions from README files in CSDs for OS/2 EE 1.1 to tune the IBMLAN.INI file. Those suggestions mainly affected a few workstation heuristic settings. Remarks are indicated by ";".

Some requester parameters that impact performance are: maxcmds, maxthreads, maxwrkcache, numcharbuf, numworkbuf, sizecharbuf, sizeworkbuf, and wrkheuristics. The network administrator may tune the IBMLAN.INI file based on the network functions primarily used by the workstation, e.g., file transfers, SQL database access, and shared serial communications. Readers should refer to the *IBM OS/2 LAN Server Network Administrator's Reference* manual, under "Advanced Information," for a more detailed elaboration on the IBMLAN.INI file and its parms.

Before the user can access the LAN, the Network Administrator must set up an OS/2 LAN Requester ID. The Network Administrator will grant the user access privileges to LAN resources, and may also update the user's Program Starter for public applications that reside on the domain controller or on another file server within the domain.

STARTUP.CMD

The STARTUP.CMD file in OS/2 resembles the AUTOEXEC.BAT file in DOS. OS/2 batch files end with the extent .CMD rather than .BAT. When the OS/2 operating system first initializes, the command processor looks for STARTUP.CMD in the root directory. If it finds the file, it executes each line sequentially. A sample workstation STARTUP.CMD file follows:

```
@ECHO OFF
START /PM FREEMEM.EXE
START /PM DIGCLOCK.EXE
START ''OS/2 WINDOWED COMMAND PROMPT'' /I
START /C /I C:\CMD\STARTLAN
START /C /I C:\CMLIB\STARTCM
EXIT
```

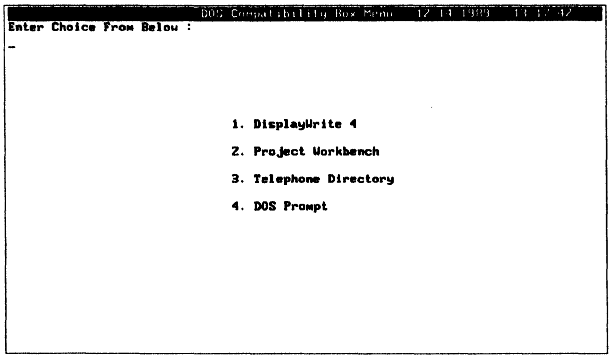
Notice the five start lines.

Booting the workstation starts FREEMEM (a free memory "high water" mark), DIGCLOCK (digital clock), an OS/2 command prompt window, the LAN Requester, the Comms Manager, and 3270 terminal emulation. The STARTLAN.CMD file initializes LAN Requester. FREEMEM and DIGCLOCK are PM applications; hence, the \PM flag is used to start them. STARTCM.CMD initializes the Comms Manager. The Comms Manager will auto-start the 3270 terminal emulation session as indicated in the Comms Manager CFG file.

The START commands embedded in STARTLAN.CMD and STARTCM.CMD are coded with parameters so that all of the processes spawned by STARTUP.CMD run as background sessions. Meanwhile, PM stays in the foreground where the individual application icons pop up, one at a time.

From a cold boot, it takes about 4.5 minutes for an 8MHz PC-AT with 6,784KB of RAM to fully initialize all of the processes started by the sample STARTUP.CMD file. A warm boot takes about 3.5 minutes. In contrast, it takes less than

Figure 9.
DOS Compatibility Box Menu



The DOS Compatibility Box Menu does not use any RAM—a limited resource for running native DOS programs on the OS/2 workstation. Yet, it gives structure to the DOS Box so users don't have to work at the DOS command prompt. The menuing system works nicely with applications on network drives as well as local programs. It's a good example of how old tools can sometimes contribute to the success of new systems technology.

1.5 minutes to warm-boot the PC-AT DOS workstation and to initialize LAN and 3270 terminal emulation.

It is not convenient to reboot the OS/2 workstation since all of its active processes must be manually shut down before rebooting. Afterward, the OS/2 reinitialization and subsequent start-up of the processes takes a while.

Applications

You can install OS/2 applications locally on the workstation and start them with the STARTUP.CMD file. The network administrator may install network applications on the OS/2 LAN Server and may grant access to users or user groups. A user may then run the network applications from any OS/2 EE 1.1 workstation on the LAN.

Access to network applications via the LAN could take the form of Public Applications. They would appear as a fourth group in Start Programs after the user logs onto the LAN if the network administrator has put the applications in the user's Program Starter profile.

If the user can't run the application under OS/2, preferably as a network application, the second best alternative is to run applications in the DOS Compatibility Box. The last and least desirable option is to shut down the OS/2 workstation and then to reboot using the DOS workstation boot floppy.

DOS Compatibility Box

The DOS Compatibility Box initializes if PROTECTONLY=NO in the CONFIG.SYS file. OS/2 EE 1.1 has one DOS Compatibility Box. Future 80386 OS/2 versions will supposedly support multiple DOS Compatibility Boxes (see Figure 9).

When the user clicks on the DOS icon, OS/2 processes stop and DOS starts—an AUTOEXEC.BAT runs first if it exists in the root directory. However, OS/2 communication programs remain active. As an example, the LAN Requester still receives messages and a 3270 terminal emulation user of PROFS will still receive PROFS notes and hear the audible signal that a note has been received. When the user hot keys back to OS/2, the DOS process stops and the OS/2 processes become active again. Note that most DOS communication programs will not run in the DOS Compatibility Box.

The main problem with the DOS Compatibility Box is lack of RAM for applications—a characteristic problem with DOS. The problem becomes more critical in the DOS Compatibility Box, however, since the OS/2 overhead reduces free RAM by quite a bit. Some configuration changes will stretch the free RAM somewhat, but an application that requires RAM approaching 450KB will definitely not fit. IBM plans to make more bytes of RAM available to the DOS Compatibility Box in future releases of OS/2.

I use a BASIC menuing system that does not take any RAM away from the DOS applications. An example of an AUTOEXEC.BAT for the DOS Compatibility Box is shown below:

```
@ECHO OFF
CLS
PATH C:\OS2;C:\OS2\SYSTEM;C:\;C:\BATL
SET COMSPEC=C:\OS2\COMMAND.COML
APPEND=C:\OS2;C:\OS2\SYSTEM;C:\;
CALL HELP ON
PROMPT $E[0;36M$P$G
BASICA C:\BAT\MENU
C:\BAT\GO
```

The menu program, MENU.BAS, writes the name of an application batch file name out to GO.BAT, which it then executes as the next step in the batch.

Figure 10.
CONFIG.SYS for NetWare Requester

```
PROTSHELL=C:\OS2\PMSHELL.EXE C:\OS2\OS2.INI C:\OS2\CMD.EXE
SET PATH=C:\CMLIB;C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;C:\;
      C:\IBMLAN\NETPROG;C:\CMD;C:\PROGRAMS;C:\NETWARE
SET DPATH=C:\CMLIB;C:\OS2;C:\OS2\SYSTEM;C:\OS2\INSTALL;
      C:\;C:\IBMLAN\NETPROG;C:\NETWARE
SET COMSPEC=C:\OS2\CMD.EXE
SET PROMPT=$I[$P]
SET SQLUSER=RPANZ
TIMESLICE=45, 125
BUFFERS=60
DISKCACHE=64
MAXWAIT=3
MEMMAN=SWAP,MOVE
PROTECTONLY=NO
SWAPPATH=C:\OS2\SYSTEM 512
THREADS=255
TRACE=OFF
SHELL=C:\OS2\COMMAND.COM /P
BREAK=OFF
FCBS=16,8
RMSIZE=640
COUNTRY=001,C:\OS2\SYSTEM\COUNTRY.SYS
DEVINFO=KBD,US,C:\OS2\KEYBOARD.DCP
CODEPAGE=437,850
DEVICE=C:\OS2\POINTDD.SYS
DEVICE=C:\OS2\MOUSEA02.SYS
DEVICE=C:\OS2\PMDD.SYS
DEVINFO=SCR,EGA,C:\OS2\VIOTBL.DCP
DEVICE=C:\OS2\EGA.SYS
DEVICE=C:\OS2\COMO1.SYS

rem The following device drivers are for access to the
rem Token-Ring Network and are inserted by the IBM LAN
rem Requester installation program. They were modified for
rem the Comms Manager configuration file (CFG=) by the
rem installer of this workstation.
DEVICE=C:\CMLIB\LANDD.SYS
DEVICE=C:\CMLIB\TRNETDD.SYS CFG=C:\CMLIB\BOBP.CFG
DEVICE=C:\CMLIB\NETBDD.SYS CFG=C:\CMLIB\BOBP.CFG
rem DEVICE=C:\IBMLAN\NETPROG\NETWKSTA.SYS /I:C:\IBMLAN

rem The following line loads the Token-Ring Network process.
RUN=C:\CMLIB\ACSTRSYS.EXE

DEVICE=C:\OS2\ANSI.SYS
IOPL=YES
LIBPATH=C:\CMLIB;C:\OS2\INSTALL;C:\OS2\DLL;C:\;C:\IBMLAN\NETLIB;
      C:\SIDEKICK;C:\NETWARE

rem start NetWare Spooler
RUN=C:\NETWARE\NWSPOOL.EXE
rem NetWare Requester files
DEVICE=C:\NETWARE\LSL.SYS
RUN=C:\NETWARE\DDAEMON.EXE
DEVICE=C:\NETWARE\TOKENEE.SYS SOURCE=1
DEVICE=C:\NETWARE\IPX.SYS
DEVICE=C:\NETWARE\NWREQ.SYS
RUN=C:\NETWARE\NWDAEMON.EXE
rem to use NetBIOS remove the rems from the following two lines
rem DEVICE=C:\NETWARE\NETBIOS.SYS
rem RUN=C:\NETWARE\NBDAEMON.EXE
rem to use SPX remove the rems from the following two lines
rem DEVICE=C:\NETWARE\SPX.SYS
rem RUN=C:\NETWARE\SPDAEMON.EXE
```

(Indented lines continue from the line above.)

After exiting the application, the batch file sequence returns the user to the menu. The menu option to go to a DOS prompt will end the batch sequence. The DOS Compatibility Box can access DOS-based servers, OS/2 servers, and NetWare servers.

NetWare Requester

Novel has released a NetWare Requester for OS/2 EE 1.1, which runs in place of the IBM LAN Requester and supports source routing across IBM Token-Ring Bridges.

In addition, the NetWare Requester comes with a TOKENEE.SYS driver that enables it to coexist nicely with the IBM Comms Manager. That means NetWare Requester, Comms Manager, and 3270 emulation can run concurrently on the same workstation using the same Token-Ring adapter.

After installing the NetWare Requester through an INSTALL.EXE program, modify the OS/2 EE 1.1 CONFIG.SYS file as in Figure 10.

The IBM LAN Requester line had to be remarked out so the NETWKSTA.SYS device driver does not conflict with the NetWare Requester. Ideally, both requesters should run together.

The SOURCE=1 parm following TOKENEE.SYS activates source routing. The NetWare Requester CONFIG.SYS displayed above uses only IPX. The NetWare Requester fileset provides additional device drivers to support Novell's NetBIOS and the SPX protocol. Those device drivers are shown in the CONFIG.SYS as remarked lines. The SET PATH=, SET DPATH=, and LIBPATH= were updated to include C:\NETWARE, where the NetWare Requester resides.

A STARTUP.CMD file for an OS/2 EE 1.1 workstation running the NetWare Requester looks like this:

```
NetWare Requester STARTUP.CMD

@ECHO OFF
START /PM FREEMEM
START /PM DIGCLOCK
START "OS/2 WINDOWED COMMAND PROMPT" /I
START "NETWARE REQUESTER FOR OS/2" /I /K /WIN
C:\CMD\NETW.CMD
START /C /I C:\CMLIB\STARTCM.CMD
EXIT
```

Notice that the NetWare Requester will run in a window (\WIN) although it is not a PM application.

Backups

The best way to back up the OS/2 workstation is under DOS. The OS/2 processes lock open files, making it virtually impossible to do a full backup while OS/2 is active. OS/2 locks some very important files, including the OS2.INI file, which contains all of the workstation configuration information for the workstation program groups. If the OS2.INI file is lost or corrupted, then all of the customized user selections are lost and the workstation returns to the defaults.

The DOS backup procedure in place, before migrating the workstation to OS/2, should continue. Tailor it for use with the DOS workstation boot floppy created as a contingency measure. Workstation backups are best done over the LAN to a mass storage device. Ironically, backup is the one workstation function that DOS still does better than OS/2.

A Final Note

After developing a model OS/2 workstation, you can copy it over the network to a mass storage device by using the LAN boot floppy designed for the DOS workstation. Workstations initialized with the OS/2 installation floppy can then download the model from mass storage via the same method. Individual workstation tailoring must be done after the download. The uploading/downloading procedure is best applied only between very similar workstation hardware platforms.

This lengthy and complex migration process takes careful planning, with provisions for fall back to DOS. And afterwards, the powerful OS/2 workstation will demand much more technical support than before. However, if you take a look at the new OS/2 applications and the potent communication functionality across enterprisewide LANs, you'll see that the rewards outweigh the price. ■

The Client/Server Iterative Rapid Development Methodology

In this report:

Datapro Summary

Client/server application development projects are often thwarted by planning and development methodologies that are inept at handling these new technologies. The techniques of a new methodology, Client/Server Iterative Rapid Development (CSIRD), are designed to accommodate the special characteristics of LAN-based and client/server computing platforms. The fundamentals of CSIRD are explored, as well as guidelines for selecting platforms and tools.

Microcomputers have the potential to dramatically change the system development process. Devoting the MIPS available on the desktop to the application development process provides the power developers need to quickly program, debug, test, and modify applications. The resulting programs have sophisticated, highly effective user interfaces. When coupled in a client-server arrangement with the sophisticated database server software now available, PCs can satisfy the security, integrity, and throughput needs of mission critical applications.

Thus, an application developed on a microcomputer LAN should be able to address a business need in far less time than a similar one developed on a minicomputer or mainframe.

Unfortunately, there are no systems development methodologies that take advantage of the unique features of microcomputer LANs. Most rapid development methodologies simply accelerate the traditional life cycle development process that originated with mainframe computers in the sixties and seventies. When such methodologies are applied on the new LAN-based and client-server platforms, the fit is so poor that the applications are often delayed or canceled. Lacking an alternative framework, most client-server development projects reinvent the wheel as the same steps are struggled over and reinvented on each project.

This report proposes a new methodology, Client-Server Iterative Rapid Development (or CSIRD, pronounced "Scissored," for short, with the intended implication of precise, clean cutting). The CSIRD Methodology takes advantage of the unique characteristics of client-server architecture. This report will explain the methodology and illustrate it by discussing how it was applied in the development of a particular application.

CSIRD delivers functionality in phases, thus delivering important business benefits very quickly. An overall design phase, overview by a system architect, and built-in feed-back loops ensure the pieces integrate into a smoothly functioning whole. The methodology also incorporates platform and tool selection—whereas most methodologies assume the existing platform will be used—and revisits that issue to be sure the project doesn't trip over an early and unfortunate choice.

This Datapro report is a reprint of "The CSIRD Methodology" by James Louderback, pp. 46-52, from *DBMS*, Volume 4, Number 7, June 1991.

The Past

Traditional life cycle systems development is dominated by the main limitation of the platform: Machine cycles are expensive. Because mainframes are so costly to use, methodologies were developed to minimize the amount of machine time used to build the system. These methodologies rely heavily upon building a complete paper-based model of the system before any coding is done. This paper-based model includes data-flow diagrams (exploded down to the *n*th level of detail), database designs, entity-relationship charts, structure charts, screen layouts, report layouts, and even pseudo code. Each of these pieces of paper requires a "walk through" review of each component by the development team.

The human mind is good at abstraction and converting symbols into tangible reality, but asking participants to visualize an entire system in their heads is a bit much. This is particularly true for end users, who often participate in walk-throughs but find it very difficult to visualize the system from paper models. In many cases, a customer signs off on a model without understanding how it will translate into reality.

Traditional project development is also characterized by large teams of specialists, each coming in to do his or her phase and then leaving. The business analysts come in to do the functional design. The senior technical analysts come in next, turning the functions defined in the first step into structure charts, pseudo code, and the database or file design. With the analysis phase finished, the programmers and senior computer scientists descend to develop computer code based on the paper model the analysts have left behind. Finally, the systems testing and documentation specialists come in and take the system from raw code into production. Except for the project manager(s) and the poor users, there is no continuity between phases. There is insufficient team cohesiveness to proceed efficiently. If a programmer has a question on a subroutine specification, he or she has to track down the analyst, who may be working on another system in another state or country. End users are never effectively drawn into the process.

PC/LAN-Based Systems

Microcomputer CPU time is cheap—once you buy the PC, the CPU cycles are practically free. Consequently, a great deal of power can be devoted to the responsiveness of the visual display, so better user interfaces can be created. More power can also be devoted to automating the development process. The result has been easy-to-use development environments that can be used for programming, debugging, and executing user-friendly systems in a matter of days rather than weeks. The resulting programs are easy to change on the fly, sometimes even by users.

However, the proliferation of all these easy-to-use development tools leads to a number of problems. It is not uncommon for an organization to have financial systems written in Xbase, personnel systems written in Paradox, and marketing systems written in DataEase. These incompatible systems cannot share data, and the programs developed with them, are for the most part undocumented and difficult to change. Developers often customize the programming languages beyond their core commands. Using user-defined functions and calls to compiled C or assembly language routines, they create what amounts to a completely new programming language familiar only to the developers, whose place in the organization is thus secure (some would say too secure).

The data security and integrity of LAN-based systems also has drawbacks. Most network-based systems use the file server as a shared hard drive, with all processing occurring on the user PC. The lack of intelligent processing on the file server makes it difficult to centralize the enforcement of data integrity or to incorporate the transaction processing and disaster recovery mechanisms common to mainframes and minicomputers. A hardware or power failure can wipe out a database file, or corrupt a catalog or index. In the event of such a failure, the only recourse is to go back to the most recent backup. And the lack of centralized integrity controls means that even if backups are done on a regular basis, there is no guarantee that the data contained in the backup is not corrupt. Users of LAN based systems are living on borrowed time.

Database Server Systems

Client-server based computing promises to change all that. Imbuing the repository of data with intelligence allows a properly designed database server to incorporate all the data integrity and disaster recovery features required for high-transaction, mission-critical systems. Centralized database servers also allow the user departments to run a variety of front ends—from spreadsheets to executive information systems to more typical forms-oriented database development tools—against commonly accessible data. Incompatible islands of data don't become a problem and data integrity protection that might be lacking in the front end can be provided centrally in the database server.

But client-server systems are inherently more complex than file-server systems. Two disparate programs have to work in tandem, and there are many more decisions to make about separating data and processing between the client PCs and the database server. Many pieces of software and hardware have to be understood and made to interact flawlessly. The systems tend to be larger and more complex than what is normally attempted on a microcomputer platform. The ad hoc approach that typifies most file server-based application development breaks down.

As a result, many organizations have reverted to traditional life cycle methodology for client-server development, which can cope with the complexity but only at the cost of slowing development. The projects begin with high expectations, fail to deliver results as quickly as expected, and political factors then rise up to derail the project. As is usual in traditional life cycle development, end users are not well-integrated into the process and their needs are only roughly understood and rarely satisfied. The result has been a decrease in the number of client-server success stories—a critical factor in the acceptance of this new technology.

What is needed is a new methodology built around the right team members working in a close-knit fashion. The methodology must incorporate design steps of a traditional development methodology, while also taking advantage of the interactive, rapid development potential of the tools available on microcomputers.

Team Composition

The first step in the process is to select the development group. A small and highly motivated group will reduce the number and duration of design and development review meetings, and prove much easier to manage than a development staff resembling a hydra.

The project sponsor is a critical member of this team. This person has to have a clear vision of the system and a clear mandate from management to spend the time and money to develop it. The manager of the department that will own the system makes an ideal sponsor, given a firm resolve and sufficient political power within the organization. Without a sponsor to steer and defend it, the development team may end up in political turf battles during system development. These battles will distract the team from its primary purpose, and may even force undesirable changes in the focus of the system.

Programmers chosen for the team must possess both business acumen and computer familiarity. It has often been difficult to find programmers capable of understanding business problems and working with end users. But in the last ten years computers have become just another business tool in the repertory that includes calculators and telephones, making business and interpersonal skills essential to programmers. This is particularly true for our iterative design methodology, which requires a single person or small team to do every step of module development. A computer scientist who can write the best C code in the world but can't communicate well with users is unsuited for this methodology, as is a business analyst who is afraid of a keyboard.

Once you've located programmers with the required savvy, separate them into two subteams, each responsible for the complete design, development, testing, and integration of a complete systems module.

If the project is sufficiently complex, each subteam will have two or three developers. One will work primarily on the database server, defining tables, rules, and security while the other(s) define screens, reports, and user processing. Given the luxury to divide the subteam into front- and back-end developers, assign programmers with relatively more business and user familiarity to front-end tasks while the back end developers should be stronger in technical capabilities. Of course, small projects can get by with only one developer to handle both front- and back-end development, provided the individual has both the database design and user interface experience to do the job. Even in the case of multiperson subteams, it is important that each programmer be familiar with the concepts important to the others' tasks.

It is also critical to have end users in the team. At least one user should be available full-time to help with the front end development, answer questions, and do testing. This user will be the eventual system owner, so that when the development team is done someone on site has full knowledge of the system functionality. It is not uncommon to have the end-user team member be a fully functioning member of one of the parallel design and development teams. In fact, on small projects, you could use one developer and one end user to build each development team. Make sure the user representative has the aptitude and training necessary to help with computer development. For instance, if the team is developing a front end with DataEase SQL, end users on the project team will, at a minimum, need to attend a week of DataEase SQL training.

In addition to the sponsor, developers, and at least one full-time end user, it is crucial to have an overall technical architect and a project manager. The technical architect is responsible for viewing the entire system as a whole, and ensuring that the implementation approach and tools are viable and valid for the system. The project manager ensures that deadlines are met, the functional design is on target, and coordinates the activities of the two teams. The project manager is also responsible for managing the relationship with the sponsor, and ensuring that the business goals of the system are being met. Finally, the project manager needs to commit members of the team for the life of the project, as continuity is very important. This methodology does not allow any warm body with knowledge of SQL or a 4GL to be plugged into the design team.

Figure 1.
CSIRD Methodology Overview

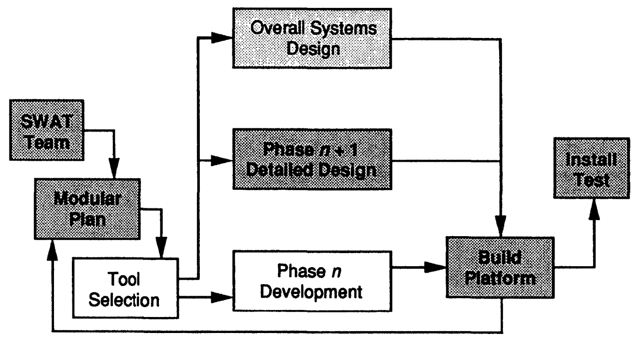


Figure 1 gives an overview of the proposed iterative development methodology. There are several essential elements:

- Initial systems conceptualization;
- Bounded discrete development;
- Parallel design and implementation;
- Continual tool and platform evaluation;
- Postponed platform selection and installation.

Initial Conceptualization

Any methodology characterized by rapid development and implementation is subject to the criticism (such as, "Where's the design phase?"). In the CSIRD methodology, an initial overall design is fine-tuned via a phased design approach. Implementation of one module will proceed while the succeeding module is being designed, all guided by the overall architect and the initial functional design. Completed modules may cause design changes to ripple back through what has already been done. In other words, parallel design and development teams work together to bring each piece of the system smoothly online. Thus, functionality that could be crucial to the business is delivered as soon as possible.

The first step after the team is formed is a concentrated three to five days spent outlining the system. This "SWAT Team" phase provides an idea of the proper spirit of the thing—to quickly and efficiently knock off an accelerated high-level functional design. SWAT Team activity is best performed at an off-site location, away from the day-to-day demands and interruptions of the workplace. The group performs problem definition, functional decomposition, and wish-listing. The result is a high-level functional design and sponsor internalization of the system vision. In addition, the first module is identified and bounded, and an overall development plan agreed upon.

The SWAT Team starts out with an initial problem definition. This is where the project sponsor details the business problem or opportunity and describes the types of functions needed to address that problem. It is important that no technical limitations be imposed on any of the functions discussed. The team should encourage the project sponsor and the user representatives to give a wish-list of the characteristics of the perfect system.

Next, the entire team tries to break the system down into logical sections. This functional decomposition is best done by writing each section down on a separate, chart-size piece of paper and assigning high-level bullet points to key elements within each function. The papers should then be tacked up on the wall, so all participants can look around the room and see the system taking shape. Some discussion of technical feasibility is necessary to bound each of the functions. At the end of this process it should be possible for the project sponsor to look around the room at each of the functions posted on the wall and be able to visualize and internalize the system and all its functions.

After all the functions have been defined, they are ranked by the developers and project sponsor. Each function is rated from most to least useful. Also, at this time some kind of "reality check" is done on each function to determine how practical it is in terms of business needs and feasibility. Once functional decomposition is complete, the team isolates and bounds the first module to be developed and separates the remaining functions into tentative subsequent modules.

Because only the first phase goes through a detailed design at this point in the process, it is important not to put too much emphasis on the components of each additional phase. One of the strengths of this methodology is that the functional design and the modules to be developed can be quickly changed as the business environment changes. By the time the first module is in development, the world will most likely have changed.

A direct marketing system my associates and I developed exemplifies the strengths of the methodology. Our client had a small outbound telemarketing operation and lots of paper. The effort was very successful, but management wanted to grow the telemarketing staff to 30 or 40 people. It was clear that a system was required to support the telemarketing and fulfillment function. So we assembled a team that included our project sponsor, a technical architect, a project manager, and a direct marketing expert. For the first day we let our client describe the type of system she wanted. As each different function of the system became clear, we summarized it on a flip chart and tacked it up on the wall.

By the end of the second day, all system functions were on paper hanging on the wall around the conference room. Our project sponsor stood in the middle of the room, looked around, and said "That's the system I want." The functions included telemarketing, fulfillment, custom letter and signature generation, cost accounting, call tracking, call forwarding, automatic outbound call routing, performance tracking, and automated data feeds.

With definition of the desired overall system behind us, we proceeded to hammer out the scope of the first phase: telemarketing and a rudimentary fulfillment system. Our tentative second phase included full-fledged fulfillment, custom letter generation, and automated data feeds. The rest of the functions were left for future modules.

Architect and Project Manager

The technical architect is the overall arbiter of how the first module and the ones that follow will fit together; he or she understands how the entire system will work together technically. The technical architect is responsible for reviewing database design and schema changes, ensuring the validity of the hardware platform, deciding on the appropriate tools for each function, and coordinating the work between the two teams. On small projects with one- or two-member teams, the technical architect can be a part-time role. On larger projects the technical architect ought to be fully committed to the project.

The process of building the development team, defining the roles of the developers, executing a SWAT Team rapid functional design, producing a detailed design for the first module, and selecting an initial development tool should take no longer than six weeks, and is usually accomplished in three weeks or less. The project manager's role here is to schedule deliverables and act as an ongoing reality check to the technical architect. As they arise, the project manager also resolves disputes between teams or among team members. Like the technical architect, the project manager can be part time on small projects. For larger projects this role should be full-time.

The First Module

The first module you decide to build should not be too large. In the example, we needed to get a system that could support 20 to 30 telemarketers up and running within two months. As a rule, development of the first module should take no more than a month or two, so try not to front load it with most of the functions. If it is impossible to break the system up into discrete implementation modules, then this system should probably be developed using an accelerated traditional development methodology.

After bounding the first module, the first development subteam does a detailed design of it in preparation for implementation. The second development subteam should also be working on the design of the initial module, but also on a high-level data model and program design of the entire project, with special emphasis on the second module to be developed. This subteam also prepares the functional specification document, which, along with the first phase detailed design and implementation plan, is the primary deliverable product of this first activity.

The functional specification document details all of the functions identified by the SWAT Team, along with their rank ordering and assigned implementation module. It also includes a high-level data model.

In the design of the telemarketing system, we pulled the development teams together and decided on a general database design. Customer tables, call tables, order tables, and others were all normalized and defined. Because this database design would have to support future functionality, such as performance and cost tracking, we allowed for those tables to be added in later. Once we had a general database structure, the first team started detailed design and development, while the second team started selecting and building the application development platform and selecting the development tools.

Choosing Platforms and Tools

Other systems development methodologies tend to discount platform selection and implementation tasks. They usually assume it already exists. If a company has an IBM 3090 and an application to be developed, in most cases that application will run on the 3090. With LAN and client server-based development, either the platform doesn't exist at all, or it will need to be reviewed and enhanced. This gives the development team a lot of flexibility, but it also increases the risk.

It is not necessary for development of the first module to lock in the final delivery platform, or even the final tools. The platform and tool selection activity is continually revisited during the iterative design and development process.

Database Server

The only primary tool locked at this stage is the database server. The database server must be easily scalable and of production quality. It should be able to run on a variety of hardware, from the PC that you will use for development to (if necessary) a supermini or mainframe if you think the data volume will require it immediately or in the future. For our telemarketing system we initially set up a small development environment with cheap twisted pair wiring, an inexpensive 10BaseT 8-port concentrator, a 386 file and database server, and some 386 workstations. As the system was being developed the offices were wired for 10BaseT, and a serious concentrator was built in a wiring closet. When the system went live we secured a more powerful database machine, gave the old one to one of the power users, plugged new telemarketing workstations into the new network, and recycled the things that we didn't need back to the development team.

Client Tools

The client development tool also needs to be of production quality. Avoid the temptation to use a front end tool that has just been announced or is in beta unless there are no other suitable tools. And if there aren't any other suitable tools, you may need to select another database server. The client development tool should allow you to quickly develop and modify the application.

Whatever you do, don't select a third-generation language (3GL). You want to be able to rapidly develop each module and easily modify it based on end-user and overall architect feedback as it begins to take shape. You may need to go back and rework parts of the module in a lower level language later for performance, but the initial front-end development should be done with a high-level tool.

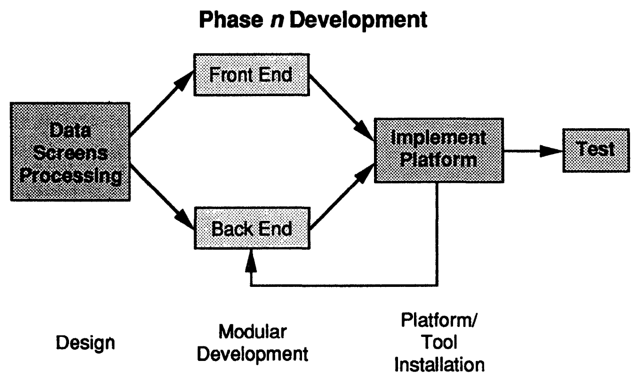
NOS

The network operating system for the development environment should be purchased with an eye towards supporting client server-systems and, if it's a foreseeable need, connectivity with minicomputers and mainframes and multiprocessing locations. You must be sure the database server and client software are tested and approved for that network operating system—relatively minor incompatibilities can have a significant impact on your progress.

Machines

The database server machine should be just fast enough to not interfere with development. Don't go out and buy the production machine now. You don't want to spend all your money on a 486-based Super Server, only to find out

Figure 2.
Parallel Development and Design



This diagram illustrates the development cycle of one application module, in this case, the data-entry screens and processing portions of a client-server application. Development of one module occurs while the design of the subsequent module is taking place. Parallel development and design ensures that pieces of the system are in place and addressing the business problem relatively rapidly, while prior and subsequent phases are being revisited and designed respectively. One group of developers is given complete responsibility for each module, ensuring consistency and completeness.

at implementation that a RISC-based UNIX server is required. Make sure that the server machine can be used as a workstation down the road if an upgrade is required. The client development machines should be expandable enough to support DOS, OS/2, or UNIX depending on the client software selected.

Parallel Development and Design

The key to this methodology is that development on one module occurs while the design of the subsequent module is taking place. Parallel development and design ensures that pieces of the system are in place and addressing business problems relatively rapidly, while prior and subsequent phases are being revisited and designed respectively. One group of developers is given complete responsibility for each module, ensuring consistency and completeness. The concept has been proven by auto makers who have improved quality by giving a group of assembly line workers complete responsibility for a particular car, start to finish. The resulting feeling of ownership and pride is a big plus in software development as well, as is the resultant understanding of how all the pieces of the module interact.

Figure 2 shows the complete cycle for development of a module n. It is important to note that while module n is being designed, module $n-1$ is being implemented. And while module n is being implemented, module $n+1$ is being designed. In addition, while this module is being designed and developed, the project manager and technical architect are monitoring and updating the high-level functional and systems design developed by the SWAT Team. These modifications occur as the business environment changes, allowing the software to adapt to the real world.

The flexibility of this approach cannot be overstated. During the development of the second phase of our telemarketing system we had one team developing the custom letter, fulfillment, and data input functions, while the other team was designing the cost tracking, telemarketer performance, and call tracking module. But because our first phase was so successful, our client was suddenly given responsibility for inbound technical calls. Suddenly we had to accommodate not only outbound telemarketers making calls and placing orders, but also the handling of inbound calls and orders. We were able to easily postpone the design of phase three, and shift that phase's team into an accelerated design and development of the newly conceived inbound module. We were able to develop a rudimentary inbound system in a month and fully integrate it into the whole system within three months. If we had been using a traditional development methodology we probably would have had to finish our existing system before even thinking about any changes or enhancements.

The Design Phase

This phase starts off with enhancements to the existing data model developed for prior phases (if any). This design is done by both the front- and back-end developers, with assistance available from the other development team and the technical architect. Complete screen and report layouts are not necessary, as the development tool chosen should make these easy to produce and change during development. The focus should be on revisions to the database design, changes required to existing programs and screens (perhaps to accommodate a new submenu structure), and any new business and data rules. New tables might be added to accommodate custom letters, for example, or new columns might be added to existing tables to add an accounting department code to an existing department table. If you make good use of views in your database design and implementation, these changes to the physical tables should not affect your existing screens and reports.

The design phase should not last longer than two to four weeks.

Once the front- and back-end developers have agreed upon the new database design, they split off and design their front and back ends. The back-end developer works on the physical tables, rules, defaults, and data types, and he or she also needs to make sure that business rules and integrity features already programmed into the database are modified to reflect the new functionality of this module. This database design might incorporate data structures from other sources around the company, and should locate or replicate data appropriately.

As the back-end developer works on the physical database design, the front-end developer works on the logical design. Existing and new views are identified, using both the database server and client software. This task also identifies new screens and reports required, and any changes to the existing front-end software. Any new processing or changes to old programs are identified and designed as well.

Processing design is done by both subteams in order to determine the best place to put new programs and balance the load on the clients and the server. Functions primarily concerned with data display and entry via screens and reports are normally placed on the client, while functions that involve data retrieval, storage, and transformation are located on the server. The subteams also address any data synchronization required between the client, the server, and any remote data stores.

Once the design is completed, development proceeds. The back end developer has to create new tables on the server and change old tables as required before work can proceed. The front-end developer proceeds with screens, reports, and client processing while the back-end developer concentrates on developing data integrity, business rules, and procedures on the server. Both teams' members should be flexible enough to help the other out should one finish first. The development process should take no more than two months.

Tool and Platform Review

Once the module has been implemented, the front-end tools and the platform are reviewed. A decision is made as to the fit between the module and tools. If the tool used to develop the front end is too slow or cumbersome, some or all of it can be rewritten in a faster or more flexible language. This should be a relatively straightforward process, because the module is already designed and developed. Client-server computing allows for client software independence, so functionality can be developed on different client programs and still be integrated into the application.

For instance, in our telemarketing system it was crucial to allow a user to bring up a standard letter and individualize it on the computer. We also needed to combine a scanned signature into the letter. Our selected front-end tool could do these things, but not easily. In a future phase we intend on rewriting the letter component in C and running it under Windows or PM along with the regular front-end application.

If the database server hardware is becoming a bottleneck, a decision needs to be made about upgrading the hardware. If the network is too slow, that choice needs to be examined and individual components replaced. Our client on the telemarketing system has already replaced the initial 80386 database server with a machine based upon the 80486. We have already upgraded the standard workstation twice by adding RAM and a hard disk to take advantage of new technology and modules. When network performance began to degrade, we upgraded the network interface card in the server with one of the new fast 32-bit Ethernet variety.

Tune, Then on to N+2

Hardware and software tuning and unit testing happens once the hardware and software have been finalized. At this point, the production environment is also upgraded or replaced. After parallel testing and training, the module goes live and the development team goes to work designing phase $n+2$.

When to Use CSIRD

The Iterative Rapid Development Methodology I've discussed has been used successfully to build many complex LAN and client-server systems. In many cases it has dramatically improved a company's competitive position by allowing for support systems to be implemented rapidly.

Many types of systems can benefit from the methodology, but some require a more traditional approach. Some systems are incapable of being decomposed into modules that can be phased into production. They may involve extremely large data sets or extremely rigid security. You couldn't use the methodology to develop an airline reservations system, an air traffic control system, an assembly line process control system, or large-scale billing systems for a telephone company.

The CSIRD methodology is most effectively used to empower professional staff rather than automate clerical functions or process control. The first wave of computer systems were concerned with automating repetitive manual procedures, but many of the strategic systems being developed today are geared more towards symbolic analysis than line functions. Examples of the types of systems that can be developed with the iterative development methodology include marketing, inventory, decision support, and customer service.

Strategic Planning Requirements

The Iterative Development Methodology also doesn't take into account the strategic value of the system being built. It is assumed that a company will do up-front analysis to determine the best place to put limited system development resources. Before committing dollars to any development, a company is encouraged to develop a long term strategic plan, and analyze the potential return on the systems development dollar when compared to other activities or systems.

Once that crucial decision has been made, the methodology will help to develop the chosen system more rapidly.

Methodology Benefits

The Client-Server Iterative Rapid Development Methodology produces quality systems sooner, maximizing return on the development dollar while accelerating system benefits. In today's changing business environment, a two-year development life cycle is simply too long for many types of systems. Marketing initiatives tend to lose force and viability over that length of time. It is far better to develop a marketing system in mere months to support a new product or service that gives a company a competitive advantage only if it is launched in a timely manner.

One of the reasons our telemarketing system example was such a success was that it was fully operational in three months, capable of supporting up to 40 users. We have subsequently enhanced it substantially, but the core of the system is still there. Changes in the business environment and in the focus of our client's department were easily accommodated in the development and design.

The flexibility of this approach cannot be overemphasized. Rather than attempting to "hit" a moving target during an extended development cycle, a system's focus can easily be modified and redirected as the business environment changes. In many cases it is better to have 20 percent of the total functionality next quarter rather than wait two years for 100 percent functionality. It is fairly easy to predict what next quarter is going to be like. It is substantially harder to divine what the business environment will be like in two years.

Another benefit of the methodology is the ability to discover doomed systems early on. One of the dirty secrets of the computer industry is that many applications that go through the design and development phases never go into production at all. Either the system's concept is flawed, or the execution is a failure. An iterative approach allows for these doomed systems to be discovered in the two to five months it takes to implement the first module rather than in the two to three years needed to develop an entire system. This saves dollars and presents the opportunity to move your staff to developing a system that will really make an impact.

The Iterative Rapid Development methodology allows you to use the interactive rapid development tools that have become popular on a PC to rapidly build secure systems on a LAN platform. You can develop a system in two to four months that has all of the data integrity and security features of a mainframe system that takes two to four years to develop. You can maximize your development dollars while minimizing the chance of failure.

The bottom line is this: Iterative Rapid Development has the potential of dramatically decreasing systems development backlog while quickly delivering quality systems that can interoperate with the rest of an organization's information system. ■

Multitasking on the 386

In this report:

This report will help you to:

- Review the key architectural features of the 80386 microprocessor.
- Know the 386 operating system and task initialization steps.
- Understand the importance of the HLT instruction in a multitasking environment.

With Intel's introduction in 1985 of the 80386, developers were faced with a processor that far exceeded any of Intel's previous offerings. In the PC marketplace, 8086 and 80286 compatibility combined with faster execution speed made the processor's success a foregone conclusion. For developers of embedded systems, however, the addition of new architectural features was even more compelling than speed.

Foremost among these feature improvements was support for 32-bit addressing. Programmers could finally get around the segmentation bottlenecks of the past and define segments as large as 4 gigabytes. For developers of operating systems, a number of new features made the 80386 a developer's dream, including a powerful new paging architecture and facilities for improved I/O management.

This Datapro report is a reprint of "Multitasking on the 386," pp. 28-39, from *Embedded Systems Programming*, Volume 3, Number 4, by Jim Betz. Copyright © 1990 by Miller Freeman Publications. Reprinted with permission.

Task Management

Taken together, the new features made it practical for designers to develop a virtual-machine operating system for an Intel processor. Before creating a multitasking virtual-machine operating system, though, designers should review the key 80386 architectural features and how they can be used to implement the basic system functions of task protection, task management, memory management, I/O device management, and interrupt or exception handing.

The 80386 provides several mechanisms to implement task protection. The operating system must allow concurrent tasks to execute as efficiently as possible without corrupting the system's kernel or each other.

The 80386 allows the task address space to be divided into memory segments, each with a base address and range. The segment is referenced by a descriptor, which is defined in the global descriptor table. A particular segment is activated by loading a segment register (CS, DS, and so on) with the descriptor number.

An operating system can use segmentation in many ways. The simplest way is to define one flat address space shared by all tasks, including the kernel, in which the base address is 0 and range is 4 gigabytes. This method is efficient, since segment register loading (an expensive operation) is minimized and segment faults are eliminated. The flat memory model is efficient but provides little protection from intertask corruption. Fortunately, the 80386 paging mechanism provides us with an excellent alternative.

The 80386 paging mechanism allows each task to have a unique segregated address space. The page directory table is a 4 kbyte array, where each 4-byte entry addresses a page table. Similarly, a page table is a 4-byte array, where each 4-byte entry addresses a 4-kbyte memory page. One page directory table and its associated page tables can define the entire 4-gigabyte address space, with each page table defining up to a 4-Mbyte region.

A page directory table and its associated tables is activated by loading the page-directory address into the 80386's control register 3 (CR3). By activating a unique set of page tables for each task, memory segregation can be achieved.

The 80386 provides a four-level privilege hierarchy. The task privilege level determines what instructions it may execute and what memory pages it may access. An attempt to access an instruction or resource outside the task privilege level will generate a general protection exception. The operating system's kernel executes at ring 0 with full access to all system resources. At the other extreme, users applications should execute at ring 3, so the operating system can maintain overall system control and integrity.

Task and Memory Management

The primary 80386 task management construct is the task state segment (TSS). The base area is essentially a cache for processor registers. Whenever a task is activated, its associated task state segment is loaded into the task register. Optionally, this segment can include an I/O permission bit map that defines which I/O ports can be access by the task.

Task dispatching decisions are obviously the operating system's responsibility. A task can be interrupted for a variety of reasons, depending on the application. It is up to the operating system's task dispatcher to diagnose the cause of the interruption and either redispatch the interrupted task or dispatch a new task.

When the 80386 paging mechanism is activated (through control register 0), linear addresses are automatically translated into physical addresses. The 80386 caches the 32 most recently referenced pages in the translation lookaside buffer to minimize table lookups. Each page-table entry has control information to record the status of the related memory page.

Access rights can be defined on a page-by-page basis. Page-accessed, page-present, and page-dirty flags can be used by the operating system's paging supervisor to implement a virtual memory mechanism. If the page-present flag is not on for the referenced page, a paging exception (INT OE) is generated by the processor. The operating system's paging supervisor can retrieve the missing page from auxiliary storage and reexecute the faulting instruction. The page-accessed flag can tell the supervisor which pages may be eligible for transfer to auxiliary storage. The page-dirty flag indicates that the page has been modified.

Interrupts and Exceptions

An interrupt refers to a signal generated by a hardware device to indicate some event and is often referred to as an IRQ. An exception signals that an illegal operation has been attempted by a task. Both events are managed through an 80386 data structure called the interrupt descriptor table. This table defines the routine to receive control when the interrupt or exception occurs. Table entries 0 through 31 are reserved for Intel-defined events.

IRQ numbers depend on how the 8259A interrupt controller base has been programmed. Since the standard 8259A base for PC hardware is eight, a conflict exists with the reserved Intel exception numbers. One of the first jobs of an 80386 protected-mode control program is to move the 8259A base to an unassigned slot. Generally, the first-level interrupt handler for a hardware IRQ will pass control to the hardware device-driver module, which is currently managing the interrupting device. Page faults are passed to a paging supervisor for resolution.

General protection exceptions are generated for a wide variety of events. It is the responsibility of the general-protection handler to determine the exact cause of the exception, which usually involves disassembling the faulting instruction. I/O instruction exceptions are usually passed to the hardware device driver that is managing the I/O port in question. Other privileged instructions are either emulated, ignored, or cause the faulting task to be aborted.

The number of exceptions generated by a task can be influenced through the I/O privilege level. This level is the minimum privilege level at which a task can issue I/O instructions. When executing at I/O privilege level 0, a privilege level 3 task will generate the maximum number of exceptions for I/O instructions and give the operating system the maximum control over the I/O environment. It is often desirable to allow well-behaved tasks to execute at I/O privilege level 3 to minimize overall system overhead.

I/O Device Management

The diversity and complexity of PC hardware makes I/O device management one of the major challenges for a virtual-machine operating system. I/O devices usually include some combination of I/O ports, interrupt lines, and memory. A special hardware device driver is usually required for each I/O device. The management of hardware devices in a virtual machine environment cuts across almost all areas of the 80386 architecture. Hardware interrupts can be fielded through the interrupt descriptor table, I/O ports can be trapped through the I/O permission bit map, and references can be fielded through the general protection handler. Hardware memory components can be managed through the paging system.

From an operating-system point of view, I/O devices can be divided into three categories: shared, floating, and exclusive. Some devices can dynamically change categories, depending on how the user wishes to use them at a given time.

For a shared device, the physical hardware is managed by the operating system, and the user tasks are presented with a logical (or virtual) image of the device. All user references to the I/O ports must be trapped and simulated. All physical IRQs must be fielded by the operating system and presented as logical interrupts to the user task after taking the logical interrupt status into account. The task may have the processor logically disabled for interrupts, or it may have the specific IRQ logically masked off in the 8259A. Some examples of this type of device are the 8259A interrupt controller and 8254 interval timer.

Floating devices have dual personalities. When a task is in the foreground, these devices may get access to the physical hardware. In the background, they have no hardware access and are simulated by the operating system. Video adapters fall into this category. When a task is visible (in the foreground), it is allowed to interact directly with the hardware to achieve optimum performance. When moved to the background, the device's entire physical state must be recorded. The device is then simulated until it is brought back to the foreground.

As you might imagine, an exclusive device is allocated to only one task at a time. Under these circumstances, it can usually be allowed to interact directly with the hardware. Parallel and serial devices can be included in this category.

In 80386 protected mode, segment registers do not contain linear addresses. Instead, they are index values in descriptor tables. How can existing 8086 applications execute as tasks in this environment? The answer is Virtual 86 mode. A task in this mode executes as if it were executing in real mode. All the 80386 mechanisms discussed previously still apply. (The exception is that the segment register values loaded by the task are interpreted in the conventional real-mode manner.) Of course the addressing limitations of the real mode environment also apply for the task.

Designing a Virtual System

Unlike the infamous 80286 real-mode implementation, you may have multiple Virtual 86 tasks running on the 80386. In a virtual-machine operating system, these tasks are executed concurrently, with each believing that is has full access to the physical hardware resources. The operating system must distribute the physical resources to the active tasks in the most efficient way possible while trying to prevent intertask corruption at the same time.

One of the first successful, virtual-machine operating systems was the VM/370 written for the IBM 370 mainframe. VM/370 has two basic components: the control program kernel and the conversational monitor system. The monitor system is basically an interactive development environment. Multiple copies can be executed concurrently to provide a multitasking and multiuser environment.

In the 80386 world, equivalents for the control program kernel and conversational monitor system have been developed. (The monitor equivalent is commonly called DOS.) By executing multiple DOS tasks in Virtual 86 mode under a 80386 control program, you can achieve multitasking of existing PC applications.

Now that we've considered some of the functions that must be performed by an 80386 operating-system kernel, it's time to take a closer look at task operation. The following is an outline of the major steps that occur in the initialization of an 80386 operating system and its associated tasks.

386 Operating System and Task Initialization Steps.

Operating System Initialization

1. Build the operating system's global data structures.
2. Build GDT, IDT, and BASE page tables.
3. Load kernel modules.
4. Load hardware device drivers.

Task Creation

1. Build the operating system's task data structures.
2. Build TSS.
3. Allocate task-specific memory.
4. Build task-specific page tables.
5. Allocate task-virtual hardware devices.

Task Boot

1. Initialize task memory (interrupt vectors, BIOS areas, and so on).
2. Initialize task-virtual hardware devices to boot status.
3. Initialize task execution (usually INT 19 to boot under DOS).

A task will continue to execute until one of four events take place:

- The time slice expires. In this case, the next highest priority task will be dispatched.
- An external interrupt occurs. If the interrupt is for a higher priority task, the currently executing task will be preempted.
- An 80386 exception occurs. The faulting instruction is either emulated or ignored, and the current task is redispatched.
- A HLT instruction is executed, which produces an 80386 exception.

Task Execution

The HLT is a very valuable instruction in a multitasking environment. It allows a task to voluntarily give up control until an external interrupt occurs. Many DOS applications needlessly waste processor resources looping on a condition that can't be satisfied until an external interrupt occurs. By issuing the HLT instruction, the task can free the processor to service other tasks until the external interrupt occurs. A simple example of this capability is:

```
LABEL1: :WAIT FOR A KEYSTROKE WITH EXTREME
PREJUDICE

MOV AH.01          KEYSTROKE AVAILABLE?
INT16H             :..
JZ LABEL1          :..NO - KEEP LOOKING

LABEL2: :WAIT FOR A KEYSTROKE WITH CONSIDERATION

MOVE AH.01         :KEYSTROKE AVAILABLE?
INT 16H            :..
JNZ END_LABEL2     :..YES - EXIT
HLT                :..NO - WAIT FOR INTERRUPT
JMP LABEL2         :  CHECK FOR KEYSTROKE

END_LABEL2:
```

A final option is task suspension by a system resources manager. This manager monitors the system for special conditions including idle tasks. An idle task could be defined as one that executes for a long period of time without doing anything, such as LABEL1. Clearly, this area requires some judgment and is beyond the scope of this report. Resource-manager, decision-making parameters should be configurable on a task-by-task basis to account for variant application-execution patterns.

Additional Considerations

A primary concern of any multitasking operating system is file-system integrity. Managing concurrent access to data files and directories is dependent on the specifications and structure of the individual file system and independent of the host hardware architecture.

We have discussed multitasking DOS applications using the 80386 Virtual 86 feature. What about breaking out of the 640-kbyte limit with applications that can execute full 32-bit addressability? The technology exists, but the industry standards have not been written. DOS applications have already been written that allow 32-bit addressability. These applications execute under a protected mode shell, commonly called a DOS extender.

Unfortunately, DOS extenders cannot coexist with 80386 control programs. The de facto standard, VCPI, that was developed to address simultaneous, protected-mode processes did not take 80386 control programs into account and is lacking in some key areas. These applications execute at privilege level 0, which means that a DOS extender can preempt and possibly corrupt the operating system's control program.

In addition, I/O port trapping for simulation and redirection of hardware devices is not possible for privilege level 0 applications. It is not possible to develop any kind of robust multitasking environment under such conditions. Multiuser systems, which require I/O-request redirection, are impractical. Help may be on the way, however. Microsoft Corp. has developed a standard called DPMI that addresses these issues. Hopefully, DPMI-based software will be under development soon.

80486 machines are now available and offer some additional advantages. The number of cycles required to execute most processor instructions has been cut in half. In addition, several new instructions have been added, including compare-and-exchange (CMPXCHG). This instruction is similar to the IBM 370 compare-and-swap instruction and can be used to facilitate tightly coupled multiprocessor support.

Current serial hardware technology was definitely not designed for modern multitasking environments. The transfer of each data byte can only be accomplished with a great deal of software overhead. In addition, current serial hardware and software is not very tolerant of the timing distortion introduced by a time-sliced multitasking environment. It is encouraging to see that the recently announced Hayes ESI specification has recognized this problem. Hopefully, it will be given consideration in future PC hardware and software environments.

The merging of 80386/80486 hardware technology with multitasking and multiuser control programs can bring a whole new level of power and flexibility to PC users. The PC industry is going through a rapidly accelerating transformation because of this technology. For better or worse, the single-machine-single-task approach to PC development must give way to a broader solution. The need for robust and far-sighted hardware and software standards is critical if we are going to realize this technology's potential. ■

Programming on the 386

In this report:

This report will help you to:

- Understand the four modes of the 386 microprocessor and what they mean to a programmer.
- Evaluate features of some of the available 386 programming tools.
- Review the high-level programming languages for the 386 microprocessor.

The 386 microprocessor provides programming power that professionals have used to craft exciting new operating systems and applications. Users can receive the benefits of the 386 chip by buying these products and using them wisely. Increasingly, though, users want the freedom to customize or integrate their commercially acquired products themselves. This requires the use of the same 386 PC programming tools that the professionals use.

While this might seem forbiddingly technical, it need not be. Certainly, assembly language programming for the 386 chip—the method that offers the greatest control over the system—requires detailed technical knowledge. But many users today know their PC systems in great detail, and for them assembly language programming is not out of the question.

For less technically adept users, the 386 system environment today sports a wide array of tools, one of which is sure to match the needs of any individual committed to the idea of personal programming.

One of the best aspects of all 386 PC programming products is that, in addition to simply developing code that takes advantage of the 386 chip's capabilities, they use the chip themselves to produce development environments of remarkable safety and comprehensibility. Using 386 PC programming products, which are far more complex than

---

This Datapro report is a reprint of Chapter 13, "Programming 386 PCs", pp. 279-291 from *Intel's Official Guide to 386 Computing* by Michael Edelhart. Copyright © 1991 by McGraw-Hill, Inc. Reprinted with permission.

their 8088 and 80286 forebears, is generally much easier and more enjoyable.

In this report, you'll learn about 386 PC programming products from the chip level outward—from assembly language through high-level languages, to operating system tools and application builders.

The Programming Environment for 386 Systems

All 386 CPU-based programming products take advantage of a rich programming environment established by the 386 microprocessor. To understand how to get the most from any 386 PC tool, you'll need at least a general idea of what the 386 processor provides programmers.

Four Modes

First, a 386 PC programmer needs to understand that the 386 chip can be viewed as having four "personalities." Any 386 system software must be written to one of these personalities.

Real Mode

When the 386 processor is turned on or reset, it is designed to come on in Real mode. Real mode is the 386 personality that virtually replicates the 8088 chip. In Real mode, the 386 chip operates as an extremely fast 8088 chip. This Real mode capability is the reason that the 386 computer can run existing PC programs. However, none of the 386 PC's new, sophisticated capabilities comes alive in Real mode.

286 Mode

Real mode runs like an 8088 chip, but what about software that was developed for the 286 processor? The 386 processor is designed as a superset of the 286 processor. This means that the 386 chip runs all 286 PC program instructions exactly as they were originally designed. When it senses their presence, the 386 system automatically supports the 286 chip's 16-bit memory addressing and all of that chip's unique features.

The 386 system doesn't actually go into a separate mode to accomplish this; the processor's native state comes with 286 chip support built in. Keep in mind, however, that the more powerful chip can completely mimic its less powerful predecessor.

386 Mode

The 386 processor's native state is 386 mode. It is a protected environment, which means that old Real mode PC programs won't run in it. It is fully 32-bit in design, supports segmented and paged memory models, and offers large segments and all the other benefits of a 386 PC that have been described in this report. UNIX, DESQview, Windows 3.0, and the other products that bring the 386 chip's power to users work in this mode.

While Real mode and 286 mode bring compatibility to the 386 chip, 386 mode (also called protected mode) is the processor's power personality. Over time, as more software appears that takes specific advantage of the 386 chip, the 386 PC's protected mode will become the standard system platform for PCs.

Virtual 8086 Mode

One of the features of the 386 chip's protected mode can be viewed as a personality in its own right. The virtual 8086 mode is the 386 chip's way of creating "virtual" PC sessions. This allows the 386 chip itself and its operating system to run in the 32-bit protected mode, while still being able to run Real mode DOS applications. In the virtual 8086 mode, each DOS session looks to the program in it just as an original PC would. In reality, the session operates in protected mode; the 386 chip "fools" the old program about its surroundings. Since each of these sessions has access to the 32-bit protected mode's full protection mechanisms, a properly written 386 system can run many of them at once, keeping each isolated from the others. Each virtual 8086 mode session thinks it is alone.

The Importance of the Four Modes

The 386 processor's four "personalities" give the potential programmer a great deal of freedom in deciding how to create software. Real mode means that old code still works and all existing DOS and Intel architecture programming tools also work. Even if you wrote code in the oldest PC assembler or the very first copy of Turbo Pascal, the 386 chip would run it.

The virtual 8086 mode assures that any of this old-style PC software will still be able to run in the latest 32-bit operating environments. There is no need to move to a more complex or unfamiliar programming environment to create a simple program to run with more complex 386 system software; write it with the tools at hand and let the virtual 8086 mode carry it into the future.

The 386 processor's complete 286 chip support means that the many programs written to take advantage of IBM PC ATs, PS/2s, and the host of other 286 chip-based computers and their peripherals will work unmodified on the more powerful chip. None of that dBASE or Lotus or EMS-related code need be wasted.

Of course, the 32-bit protected mode environment holds the key to a host of new possibilities for PC programs.

Low-Level Programming Languages for the 386 Processor

Assembly language is the programming scheme that gets closest to the processor and its surrounding system resources. In assembly language, an extensive set of chip, memory, and input/output binary machine language operations is gathered together and assembled into commands that are comprehensible to users. These commands take some getting used to because they control computer activities, as opposed to human ones. They deal with registers, segments, protection levels, and so on. This means that assembly language programmers need to understand the system they are working with in detail. However, this assembly language knowledge translates directly into power. Assembly language provides pinpoint control and the fastest performance of any programming language.

Assembly language programming tools come in three parts. First is an editor, where the programmer enters the assembly language instructions along with comments that serve as both personal notes about what is going on and as internal documentation of the program. Next is an assembler. The assembler reads the instructions and translates them into machine language code for the specific system where they will run. The third part is a linker, which binds an assembly language program to the conventions of a particular operating system. So an assembly language program is written for a 386 system using 32-bit instructions, compiled by a 32-bit assembler, and then linked to a system that runs on a 386 PC, such as UNIX or OS/2 version 2.0.

Remember, all existing PC assembly language programs are supported by one of the modes of the 386 system. You should move to one of the products described here either to get at the full range of 32-bit protected mode power or to have a more modern, interactive environment for writing traditional PC programs.

Microsoft Macro Assembler

Microsoft's Macro Assembler (MASM) is widely accepted as the standard assembly language system for PCs. This version supports the new 32-bit instructions and memory models for the 386 processor. It has been designed to make it especially easy to link 32-bit 386 chip assembly language code with programs written in high-level languages, such as C and Pascal. The assembler can create programs that run under OS/2, DOS, or, using a facility called BIND, under both.

MASM operates with a window-oriented debugging program, called CodeView. This product can help a programmer find problems in assembly language code quickly. The latest version of CodeView supports OS/2 by allowing programs up to 128 megabytes to be debugged. It can also keep track of the multiple-threaded operations of a multitasking program.

Borland Turbo Assembler 2.0

Some programmers favor Borland's Turbo Assembler 2.0 because of the support its environment provides. The assembler runs with both a debugger and a profiler that review program code and make suggestions on how to improve its performance.

Like the Microsoft assembler, Turbo Assembler provides strong interfaces to high-level languages. Unlike MASM, it supports the full instruction set of the 486 processor. Turbo Assembler does not yet have the complete OS/2 support that Microsoft provides.

High-Level Programming Languages for the 386 Microprocessor

Assembly language is powerful, but for many people it borders on the incomprehensible. High-level languages, such as C and Pascal, offer a much more straightforward way to create programs for the 386 system. High-level languages gather a number of assembly language instructions together into commands that deal with such familiar things as characters, numbers, variables, files, screens, ports, and other computer elements. By manipulating these elements, users can direct the computer to do whatever they want. Like assembly language tools, high-level languages generally come with an editor, a compiler, and linkers. The result is a machine language set of instructions that the computer can run, called *executable code.*

The entire current generation of high-level language products supports the 386 microprocessor to one extent or another. Generally, the more powerful languages provide the strongest 386 chip support, while the more introductory languages, such as BASIC, offer the least.

The look and feel of high-level languages can vary widely, even among products that support the same language. If you don't already have a favorite environment, shop around and experiment with a number of different products until you find one that is comfortable.

The C Language

C is the programming language that UNIX was developed in, so it makes a natural complement to the UNIX operating system. C is favored heavily by professional programmers because it has many of the portability features that UNIX has. It also produces code that runs nearly as fast as assembly language.

C is a tightly structured, set-oriented language. It provides a comfortable environment for anyone relatively well schooled in math or logic and requires a fair amount of PC knowledge.

Intel has recently released an Intel 386/486 C Code Builder Kit designed for the creation of DOS programs (it doesn't yet support OS/2). This collection includes a 32-bit C compiler that is compatible with Microsoft C 6.0, a DOS extender, a debugger, a libraries linker, and utilities. The kit comes with unlimited free phone support from Intel.

Other leading C compilers for 386 PCs come from Microsoft, Borland, Metaware, and Watcom. Microsoft's compiler is large, solid, and a bit austere. Borland's is small, responsive, and extremely friendly. Metaware's compiler has a UNIX-like flavor to it. Watcom's is the least forgiving and not very friendly, but produces the tightest code.

One of the big advantages of C in 386 systems is that this language interacts smoothly with assembly language. That means a programmer can use C for most of the programming and then drop into assembly language for tricky memory or chip-related tasks.

Pascal

Pascal is the classic structured programming language. It breaks program tasks down into neat routines, something like the paragraphs in a story. This makes Pascal program code among the easiest to read of any language and makes the language much less difficult to learn. Yet, Pascal is capable of producing program code swift enough that some commercial packages are written in it.

The most popular Pascal package by far is Borland's Turbo Pascal, one of the most forgiving and instructive programming systems ever created.

Other Pascal versions with 386 chip support include Metaware's Pascal 386/486 and Microway's NDP Pascal 386. Both of these compilers offer richer sets of 386 system tools than Borland's, but they are much less friendly to use.

FORTRAN

FORTRAN is an acronym for formula translator. It is an old programming language, originally developed for IBM mainframes. FORTRAN's primary area of impact is in scientific or mathematical situations. Microsoft makes a FORTRAN compiler that works smoothly with MASM. Other major FORTRAN versions that support the 386 microprocessor come from Lahey Computer Systems, Microway, and Watcom.

Other Languages

Most programming languages other than C, Pascal, and FORTRAN have added 386 microprocessor support to the latest releases of their compilers. Most of these languages have specialized uses and wouldn't interest a nonprofessional programmer. Zortech, for instance, has a 386 system version of C++—its environment for the creation of object-oriented programs and RR Software produces 386 chip versions of ADA—the programming language used in U.S. Defense Department projects. If a 386 system-specific compiler can't be found, users familiar with a particular language should be able to create code for the 386 system by linking 32-bit assembly language routines into their code.

Systems Software Programming in the 386 CPU-Based Environment

UNIX provides many of the tools common in a programming language. This trend is continuing in all 386 PC operating environments. Windows 3.0 and OS/2 along with UNIX offer extensive *application program interfaces* (APIs), which are essentially programming language commands within the operating system. Even lowly old DOS has become the home to programming tools that allow its programs to get at some of the benefits of the 386 chip.

It has become common, in 386 systems, to make available kits that people can use to create software that uses each environment's API.

DOS Extenders for 386 Systems

DOS extenders produce programs that are compatible with DOS's Real mode, but are capable of using many of the memory management benefits of the 386 microprocessor. These products can be used by individual programmers as well. The appropriate time to turn to a DOS extender is when a user has an existing DOS program that is constrained by the 640K limitations of DOS. It is much easier, in most cases, to revise the program to include DOS extender benefits than to completely recast it for the 386 chip.

Phar Lap

Phar Lap was used by AutoCAD and Paradox to create their 386 system versions. It brings access to 4 gigabytes of flat 32-bit memory space without sacrificing DOS compatibility. Phar Lap also provides 386/VMM, a virtual memory manager that uses the paged memory scheme of the 386 chip to allow programs that are bigger than physical memory to run.

Ergo Computing

Ergo's DOS extender is included with Turbo Pascal. It opens up 16 megabytes of memory and works with DESQview, 386Max, and the Watcom, Lahey, Metaware, and Microway compilers.

Windows 3.0 Tools

Not only does Windows provide the opportunity to create graphic DOS applications, it also provides a method for building applications that blend DOS familiarity with full access to the benefits of the 386 chip.

Windows 3.0 Software Development Kit

Anyone interested in serious program creation under Windows 3.0 can purchase a Windows 3.0 Software Development Kit from Microsoft. This package includes Windows plus a huge storehouse of development tools, code samples, debugging aids, and documentation. The kit is designed to interact with standard C compilers.

Windows Batch Languages

Windows does not provide a batch command processor apart from the traditional DOS batch language. Two third-party developers, however, have produced powerful command file builders that go far beyond what DOS can do.

PubTech Batchworks: PubTech Batchworks offers two programming services—macros and batch programs. Macros let the user modify individual applications by adding new functions to them. PubTech macros can build dialog boxes; open and size windows; and handle match, statistical, and file control operations. PubTech batch programs extend the macros beyond individual applications. A batch file can run different programs, place them in specific windows, operate on them, and exchange information between them. PubTech batch files can also put Windows front ends on DOS applications and can even create simple applications, such as a time-triggered file backup utility.

Bridge Tool Kit: Larger, more expensive, and more complex than PubTech, the Bridge Tool Kit provides much richer application integration and automation tools. With Bridge, the user can create links between applications, structure entire subroutines in an almost Pascal-like fashion, build screen menus, and command initiation routines. In essence, this product allows users to completely customize their Windows application environments. On 386 PCs, Bridge can even route individual keystrokes to DOS applications (so that they operate under Bridge control just as they would under user control) and interact in other ways that tie older applications tightly into state-of-the-art Windows processes.

Application Builders

Windows 3.0 has spurred development of a wide array of other application development tools. To begin with, most applications written for this environment include sophisticated scripting languages that allow significant customization, automation, and integration with other programs. At the high end stand four products that bring to PCs capabilities similar to those of the Macintosh's HyperCard visual development system.

Toolbook: With Toolbook from Asymetrix, the user interacts with a "book" containing "pages." Each page represents a screen presentation of information derived from actions on previous pages. By linking pages in various ways, a user can create all sorts of information-related applications in Toolbook, such as an address file, a training manual, and so on. Toolbook features links to full-motion video and digital sound and offers deep support for Windows' DDE interprocess communication scheme.

Plus: From Spinnaker Software, Plus can actually load and read Macintosh HyperCard stacks. It also runs in OS/2. This makes Plus the quick development system of choice in cross-platform systems. However, in order to achieve this multiple platform goal, Plus does not support DDE, which is specific to Windows.

Guide: A hypertext authoring system from Owl International, Guide excels at linking chunks of information together. In the Windows version, these information pieces can include graphics and even video segments. Guide is considered by many experts to be the best informational application builder for PCs.

ObjectScript: Easy to use but less powerful than some of the other products, ObjectScript comes from the French firm Matesys. It lets the user simply drag predefined elements, such as buttons, boxes, tables, and so on, into place on the screen and then fill out a form that instructs the program about how each element should function. In this way, any object on the screen can be linked to a script that executes whenever the object is properly manipulated. Interestingly, ObjectScript produces two kinds of scripts—one akin to BASIC and the other more like C.

OS/2 2.0 Software Development Kit

Although OS/2 2.0, the 386 chip-specific version of OS/2, isn't due for general release until well into 1991, the operating system is available now in the form of a Software Development Kit (SDK) from either IBM or Microsoft. In addition to a prerelease version of the operating system—which supports multiple DOS sessions using the 386 chip's virtual 8086 mode, plus the processor's full set of memory options—the OS/2 2.0 SDK comes with the following:

- *Printed Documentation,* more than 2,000 pages worth, touching on every aspect of OS/2 operation and APIs
- *On-Line Documentation* in the form of extensive help screens built into OS/2 and QuickHelp, a program designed to help developers using character-based editors to build OS/2 code
- *Microsoft C,* a special version of Microsoft's C 5.1 compiler that emphasizes 32-bit operations
- *Microsoft Macro Assembler 5.1,* the 32-bit version of the standard assembly language tool for PCs
- *Debuggers,* including Microsoft's CodeView software debugging system, as well as special debuggers for OS/2 kernel code and Presentation Manager screen modules
- *Network Redirector,* software that allows PCs using the OS/2 2.0 SDK to link with existing DOS or OS/2 networks
- *Twenty Code Samples* that programmers can use for education or as part of their own software

UNIX System Tools

UNIX, too, offers the programmer a rich supply of tools for building 386 PC software. Because of its multitasking, multiuser character, UNIX presents a greater programming challenge than other 386 system environments; its programs tend to be built up in layers, with programming knowledge required at each layer. Still, this complexity carries with it the potential for crafting unique group-oriented benefits from the power of the 386 chip.

Open Desktop Development System

SCO's Open Desktop is not only a graphic environment for running UNIX and DOS applications together, it is also a development platform for creating graphics-oriented UNIX programs or for moving older UNIX programs into a graphic future. Because it is built on UNIX System V/386 version 3.2, Open Desktop comes with all of UNIX's programming tools built in. In addition to these, the Open Desktop Development System, an add-on to the basic Open Desktop environment, contains the following:

- *UNIX Source Code Control System*, which is essential for any tinkering in the UNIX environment
- *Microsoft C*, included to help users create code with an eye toward cross-development between UNIX and OS/2 or DOS
- *AT&T C*, which is used to create many traditional UNIX applications. It can be used to create new applications or as a basis for revising older UNIX programs to take advantage of new Open Desktop features
- *Microsoft's Macro Assembler*, the standard assembly language programming system for Intel architecture PCs, including 386 PCs
- *Microsoft CodeView*, a program that helps debug problems in software code
- *PCLIB*, the DOS-under-UNIX libraries utilized by Merge 386. It allows UNIX programs to be written that can use DOS files and services
- *Motif 1.0.1*, all the code required to build applications with the three-dimensional Motif screen look and feel
- *X Windows Tools*, which a programmer can use to describe graphics sessions to different kinds of screens or screen systems
- *Xsight*, development tools that bring X Windows compatibility to DOS PCs or software

DESQview/X Development System

DESQview's X Windows product offers a uniquely broad set of tools for creating X Windows applications. This system uses X Windows' ability to take a single program and, chameleon-like, shift its screen appearance from machine to machine or session to session.

In addition to the basic DESQview/X product, the development system includes the following:

- *OSF/Motif*, a toolkit for building a 3-D look into software
- *Xol*, a three-part toolkit—window manager, workspace manager, and file manager—that implements the Open Look screen interface. Xol takes up a hefty 1.5 megabytes of RAM
- *Xview*, an alternate system for creating Open Look screens. It uses Sun Microsystem's SunView set of programming conventions, which is an advantage to anyone familiar with that environment, but an impediment to a non-Sun user
- *XVT*, the most interesting of DESQview's tools. This set of utilities creates code that can use X Windows' services and produce either Open Look or Motif screens. The same code, however, can also be linked to Macintosh, Windows, or OS/2 Presentation Manager screen APIs instead

All of these DESQview tools are designed to work with standard DOS extenders and compilers. ■

Iconic Programming

In this report:

Datapro Summary

Iconic programming, also known as visual programming, offers a more direct link than traditional textual languages between an individual and a machine. This report explains how iconic languages make it easier to develop programs than text-based languages, especially for non-computer specialists. It presents an overview of the iconic programming environment, and addresses the three most common criticisms of the language.

Almost all programming languages developed to date have a text-based syntax that requires professional knowledge and experience in programming. Thus, when casual users who are specialists in particular application domains have programming requirements but are unfamiliar with computers, professional programmers are needed. However, the programmer may often fail to represent a user's requirements precisely, which makes it difficult to develop a program that satisfies the user. Non-computer specialists need to be able to program to satisfy individual requirements and expectations.

One way to attain this is through the use of visual information in programming. This kind of programming, called visual programming, is done through visual interaction with the system. Visual programming provides more direct communication between a person and a machine and makes it easier to create programs.

Visual programming where icons are used primarily in program development is called *iconic* programming. Here, the role of icons is no longer restricted to specifying commands for system management, as in a Macintosh-like interface.

Iconic programming has the potential to provide an advanced programming environment, but could it actually be a new programming paradigm? We believe iconic programming can contribute to the realization of truly user-friendly languages and environments, based on our experiences of developing the Hi-Visual iconic programming environment.

© 1990 IEEE. Reprinted, with permission, from *IEEE Software* Volume 7, Number 6, pp. 63-68; November 1990; by Tadao Ichikawa and Masahito Hirakawa.

Pioneering Work

The effectiveness of iconic programming was first demonstrated in several application domains, including CAD, simulation, and games. In a pinball construction game, for example, one system prepared a set of parts used in a pinball game. The user freely specified a layout of parts and defined pinball games as he or she liked.

After this effort, researchers brought the idea of iconic programming into general-purpose languages. One of the most well-known, pioneering works on general-purpose iconic programming is Pict, developed by Ephraim Glinert and Steven Tanimoto.[1] In Pict, statements like an assertion or a conditional branch in a conventional, text-based procedural language like Pascal are visualized as icons. You program by arranging icons on a screen and specifying the connection among them in accordance with the control flow. Figure 1 shows an example Pict program from Glinert and Tanimoto's article. Color (not shown in this reproduction) identifies variables in a program.

In first generation iconic languages, iconic schemes were used only to interface to existing languages; they were not independent languages in their own right.

Feasibility Debate

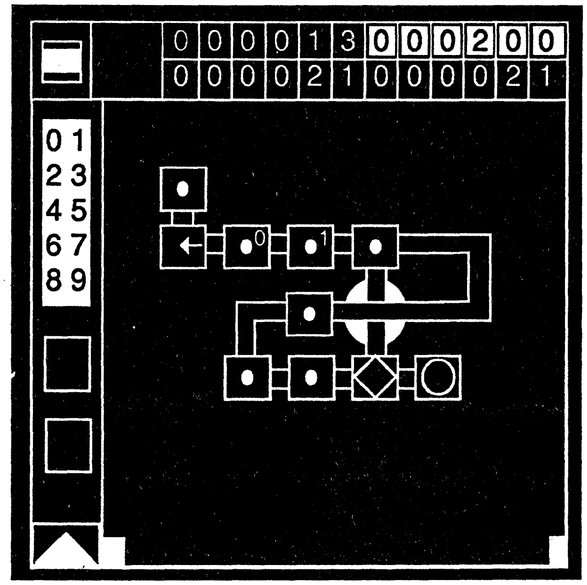
There have been many provocative experiments in iconic (visual) programming, ranging from theoretical to practical ones, covering man-machine interaction models, icon semantics, icon-image design, metrics for visual programming, visual-language grammars and parsers, and spatial layout techniques, as well as specific languages and systems.

As a basis for implementing iconic programming systems, the object-oriented model has been reasonably accepted. Moreover, several commercial products of iconic programming systems have been actually developed, including National Instruments' Labview, Metaphor Computer Systems' Metaphor, Next's Interface Builder, and Serius's Serius 89.

Some people accept the potential of iconic programming, but others seem reluctant. There are three basic criticisms:

- *Iconic programming may be good for beginners, but not for computer professionals.*

Figure 1.
Example of Pict Program



The advantage of iconic programming over textual programming is that it can make learning and understanding easier, especially for non-computer specialists. Most people agree with this assertion. But how can iconic programming benefit computer professionals? In several ways, depending on the level of the software architecture.

At the lowest level, some researchers have tried to improve the performance of conventional textual languages by applying icons to program expression. But people using these languages still must know the language model, which is very computer-oriented, and thus the user pool is restricted to computer professionals. Because computer professionals are familiar with textual programming, they do not mind using textual languages. Furthermore, mouse-based specification is sometimes annoying for computer professionals. Thus, they may prefer textual programming over iconic programming.

So is iconic programming meaningless for program development at this low level? No. Iconic programming is effective, especially for introductory courses in languages. Indeed, Glinert and Tanimoto proved through an experiment that iconic languages are *easier* to learn than textual ones.[1]

What about for higher level tasks in, for example, office information processing, simulation, computer-aided instruction, and graphical user interfaces? Imagine that you are going to produce a program in such an application domain by using a conventional textual language. Why not try to remove yourself from the text-based, precise language syntax, which is very computer-oriented? After all, what you need to pay attention to is the components or behavior of the components that you can see in an actual application environment, not the language syntax. In such programming, icons are effective because their expressive power and abstraction level is higher than that of text.[2]

- *Iconic programming may be good for small programs, but not for large programs.*

A frequent criticism of iconic programming is the wasteful use of screen space, which makes it difficult to make a large program because you can only see a small part of it at once compared to what you can see in textual programming.

Of course, screen-space usage varies, depending on the application and the level of programs to be developed. Iconic programming is actually not always worse than textual programming in screen-space usage.

Furthermore, in iconic programming, you can locate language constructs (icons) that are semantically related in a program close to each other. This seems reasonable and valuable when developing large programs.

Here's an example: Figure 2a shows a program based on Pascal syntax; Figure 2b shows the corresponding Pict-like program. In the Pascal program, statements 7 and 8 (the Else part) are placed at the bottom of the program, which is far from the associated If statement. But in the Pict-like program, these statements are placed close to the If statement. This makes it easier to understand the program. Three-dimensional display of hierarchical relationships between modules (icons/programs) will help programmers recognize the program structure even more easily.

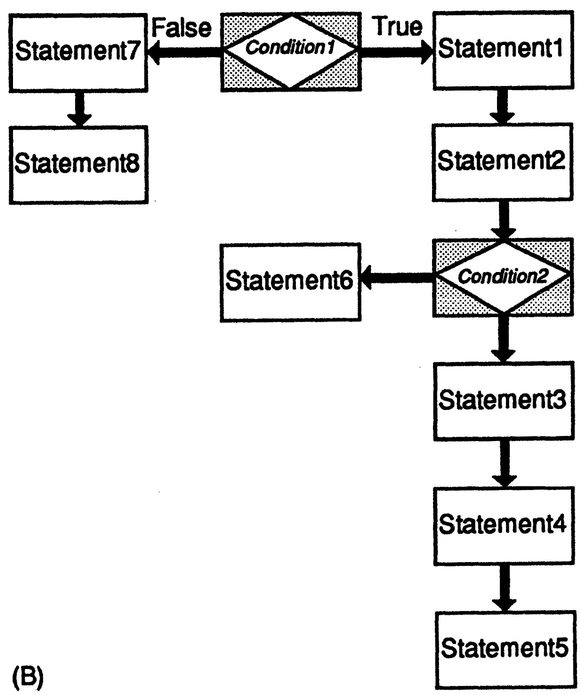
You can also register the program as an icon and use it as an element in higher level programs.

Another approach to enhance the availability of visual programming for large program development is the use of pan and zoom facilities for controlling program appearance. With such tools, you can change the program appearance from an entire structure to the smallest details of the program, and vice versa. A similar technique has been applied to drawing editors.

Figure 2.
Program in (a) Pascal Syntax and in (b) Pict-like Format

```
If Condition1=True Then
 Statement1
 Statement2
 If Condition2=True Then
  Statement3
  Statement4
  Statement5
 Else
  Statement6
Else
 Statement7
 Statement8
```

(A)



(B)

The graphical display of the latter makes relationships clearer.

These ideas may still not prove the feasibility of iconic programming for producing large programs in all application domains, but they certainly show it is possible in at least some. The most successful example of systems that can actually be applied for producing large programs is a logic design system in a large-scale-integration (1,000 to 10,000 circuits per chip) CAD environment. Such CAD systems are powerful enough to design actual LSI circuits, and in fact most designers could hardly imagine designing an LSI circuit without using it.

- *Are there really reasonable approaches to the design of good icon interfaces?*

As several researchers have pointed out, an iconic interface is not always the best choice for constructing a good user interface. Kathleen Potosnak has presented the guidelines[3] for making the best use of icons: Make icons easy to understand. Avoid misleading analogies. Don't violate population stereotypes. Use icons for appropriate purposes. Carefully design iconic interaction.

Here, the most critical problem seems to be the difficulty in designing icon images. Icon design must be done carefully so most users can intuitively and correctly understand the icons' meaning.

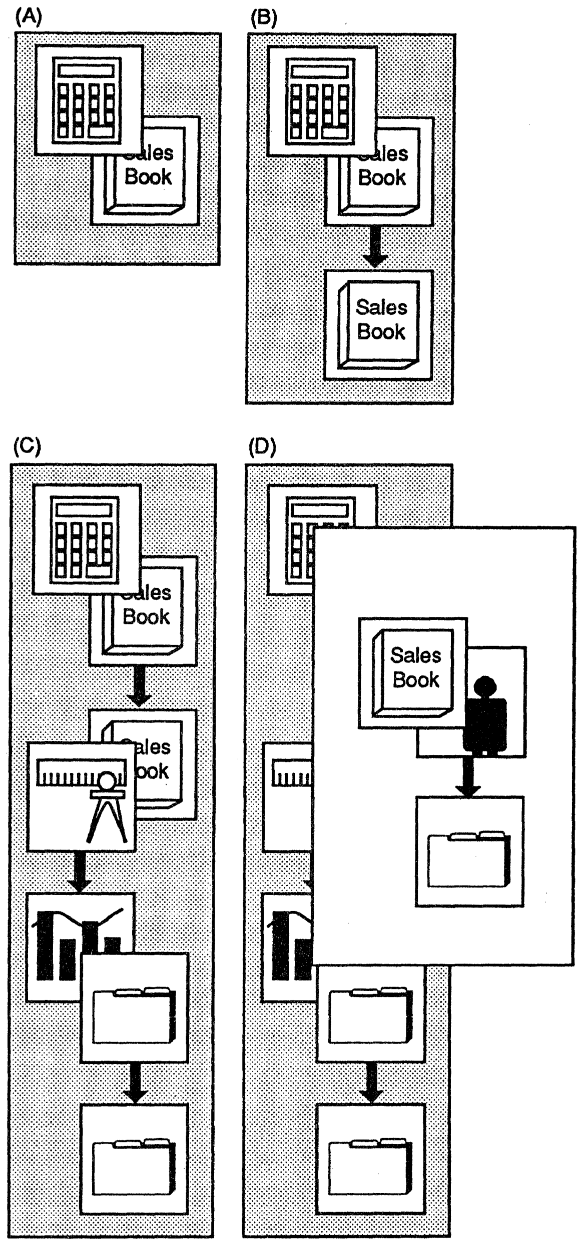
In iconic systems, icons are usually classified into two types: data and function. In general, designing icons representing data doesn't present too many problems, but it is difficult to design icons to represent functions, since functions are inherently not visual.[4]

To offer a solution to this problem, we proposed a new framework for icon management.[5] In this framework, icons representing functions are not provided. The use of icons is limited to represent actual objects (like papers, sales books, and scissors) or concepts already established (like companies and departments) in a target application environment. In this framework, you specify a function in combination with two icons: Each icon can take an active or passive role against the other. The role sharing is determined dynamically, depending on the environment in which the icons are activated.

Hi-Visual 88 is an iconic programming system based on this framework. Consider this example of how programming is done in it: Assume that you are office workers who are going to construct a program to sum up monthly sales amounts, draw a graph in accordance with the calculation result, and then keep the graph in a folder.

First, you overlap a sales-book icon and a calculator icon on a programming window, as shown in Figure 3a. To combine icons, the system infers a certain function to be executed. In this case, assume that a sales-amount summation function is associated with the icon pair. The system executes the function and generates a new icon as the result of execution (Figure 3b). You continue programming by overlapping another icon on the existing icon. Finally, you get the program in Figure 3c.

Figure 3.
Programming in Hi-Visual



(a) You first overlap a sales-book icon and a calculator icon on a programming window. (b) The system infers the sales-amount summation function and generates a new icon based on its execution. You continue this overlapping-based programming until you get (c) the desired program. (d) With its secretary icons, Hi-Visual lets you register such programs for use as procedures in higher level programs.

You can register this procedure for later use in defining higher level programs. Hi-Visual 88 provides secretary icons to support this facility. For example, assume in Figure 3c that the sales-

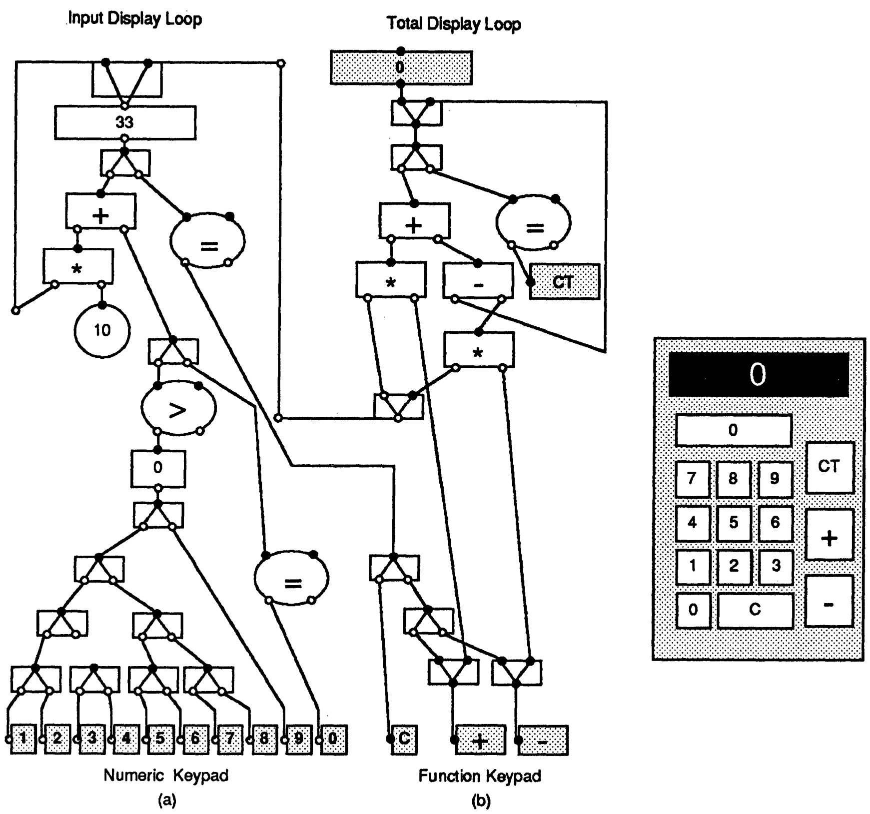


Figure 4.
Simple Calculator Program in the Language for Intelligent and Visual Environment

(a) The circuit at the left sets the value to be calculated; the + and - keys on the right add or subtract the values. (b) Resulting icon, once selected icons and all wiring are hidden from the user.

book icon and the other icon are the ones to be input and output, respectively, from the procedure. You can get a folder immediately by handing a sales book to a secretary, as shown in Figure 3d, without saying anything more about it.

Interfacing

Iconic programming was first introduced in an environment where components in the application are inherently visual, like CAD systems, computer games, and simulators. Another example of its use is in the user-interface management systems that have become available recently. In these systems, the user interface was designed so the user feels comfortable with the view. Some systems have implemented interesting ideas:

Some systems let icons have different images. The displayed image, like the shape or size, is decided when the icon is displayed in accordance with the icon's state. An example is an indicator of a meter, which changes according to the internal value associated with the meter. Another example is that, in a CAD environment, a gate (icon) can be highlighted to show that it is active.

Furthermore, icon images are not limited to two dimensions. Some recent iconic programming systems can handle three-dimensional images. These systems display 3D images of, for example, desks, cabinets, and secretaries that can be seen in an office environment. One such system, Language for Intelligent and Visual Environment,[6] also provides facilities to change the viewpoint and visualize the dynamic behavior of a program through animation, which makes program development and debugging easier.

Another technique to achieve reality in interfacing is realizing that the program's (internal) view may differ from its presentation (external) view. For example, in Intercons,[7] you specify a program by arranging data and function icons and defining wires to connect certain icons with associated icons first. After specifying the program view, you define a presentation view of the program.

As an example, Figure 4a shows the simple calculator program used in the paper on LIVE. The circuit on the left side sets the value to be calculated, where the lower left quadrant contains the buttons (numeric keypads) that output the corresponding value. Pressing the + or - key on the right side adds or subtracts the input value to or from the value (which is the result of the previous calculation) in the display at the top right.

Once a program is built, you can hide all but the selected icons (and all wiring) from the user. You can then resize and reposition the visible icons to form a presentation view, as shown in Figure 4b.

Conceptual Activities

The work on interfacing has demonstrated the effectiveness of iconic programming. But is it enough? To further enhance the feasibility of iconic programming, there must be programming methodologies that are more effective in an iconic environment than in a textual environment.

A technique that looks most promising is programming by example.[8] In this technique, specific data values and objects are regarded as examples to be processed in a program, and the programmer demonstrates the program behavior specifically in response to those input data values and objects. Example-based programming could help users construct a program because it would let them proceed with program development step by step along the actual input values. Using examples as tools for teaching is something we are all familiar with.

An example programming-by-example system is Metamouse.[9] In it, all possible patterns of input data may not necessarily be indicated. The system then infers the general program structure from the limited examples. Artificial-intelligence researchers call this scheme "automatic programming."

(Systems requiring the user to specify everything about a program are called programming with example, but they may also be called programming by example in the broad sense. Programming with example is characterized as "Do what I did," while programming by example might be interpreted as "Do what I mean."[8] The term "programming by example" includes both inferencing systems and programming-with-example systems.)

Programming by example does not necessarily have to be iconic programming. But iconic programming would contribute to the success in developing programming by example systems.

For example, suppose you are applying a value to a function module, where the value and function module are also visualized as icons on a screen. It would certainly be more enjoyable if you can specify your demand by dragging the value icon onto the function icon.

Another important concept that might work effectively in an iconic programming environment is metaphor. Unfamiliar concepts of programming are mapped onto familiar, real-world concepts that can be easily understood. The desktop metaphor in a Macintosh-like user interface is the most well-known example, but it doesn't work for programming. An example of a metaphor applied to successful programming is a theater metaphor in the Rehearsal system.[10]

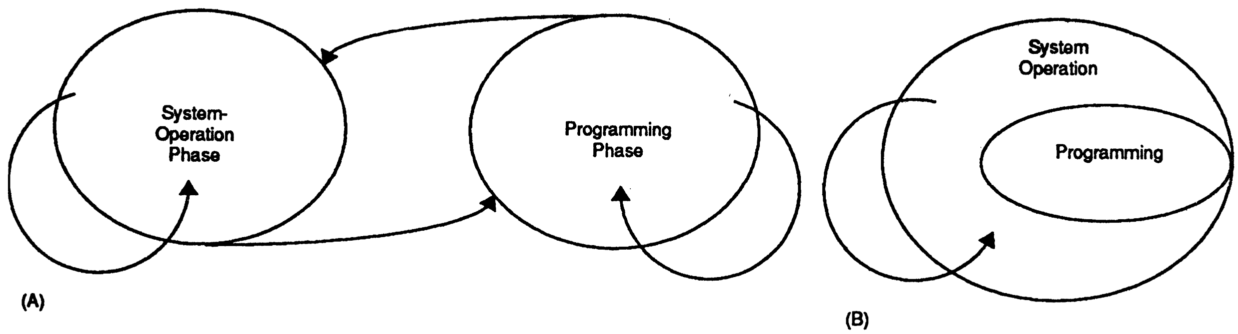
In the Rehearsal world, which is very object-oriented, programming is modeled the same way as creating a theatrical production. The basic components of the production are performers (objects). These performers interact with one another on a stage by sending cues (messages). Programming is carried out by first auditioning the available performers by sending cues and observing their responses to determine which are appropriate for the planned production. Next, the chosen performers are copied and placed on a stage. Production rehearsal begins by showing each performer what actions it should take in response either to the user's input or to cues sent by other performers. You can store the resulting production for later retrieval and execution.

More on Reality

Progress in iconic programming research has been moving toward the accommodation of reality in program-development activities. To further improve the reality of iconic programming, the following changes must occur:

- In existing computers, programming mode and program-execution mode are definitely separated, and the system changes its state from one to another during operation, as Figure 5 shows. Separating the programming mode from the program execution mode seems to be very computer-oriented. In a future computing environment, facilities to build up a program automatically from man-machine communication (that is, modeless) will certainly be required (see Figure 5b).
- Conventional pointing devices, like a mouse, are not satisfactory for advanced iconic programming, so we need an alternative pointing device. Here's a simple example of why we need such a device: When several icons (or windows)

Figure 5.
Relationship of Programming Mode and Program Execution Mode



(a) Current systems separate the programming mode and program-execution mode. (b) Future systems will be modeless, where programs are built automatically from man-machine interaction.

are overlapped, it is not possible to move, for example, the top icon to an arbitrary place in the icon stack, since the conventional mouse cannot indicate a certain point along the depth axis. Window systems that can control 3D space are also needed.

Of course, in addition to the engineering approaches studies from the scientific/artistic point of view—like a cognitive science—is essential for the development of a truly user-friendly iconic programming environment.

Conclusion

To realize a truly user-friendly iconic programming environment, we should reduce the gap in communication capacities between person and computer. Users shouldn't be forced to learn computer-oriented concepts and mechanisms. One guideline for developing such a friendly programming environment is that users should *enjoy* communicating with the computer.

Most research in iconic programming has been directed toward the development of particular languages and systems. But basic research in iconic programming is also required. Research areas include icon semantics and formal specification of iconic languages.

In addition, iconic programming is not just the visualization of program primitives under existing programming paradigms. Iconic programming might be able to provide a *new* paradigm for programming that could drastically change programming style and thus the availability of computers for users.

However, iconic programming is not a panacea. Icons are abstract, instantly comprehensible, and universal, while text is rational, analytic, and precise. Iconic programming and textual programming should thus be *complementary* to each other.

Iconic programming, in concert with textual programming, will certainly provide a more friendly programming environment for computer professionals and casual users alike. Even professionals in specific application domains who need not be programmers will be able to make programs by themselves, resulting in an as-yet unpredictable improvement of software quality as well as productivity.

References

[1]E.P. Glinert and S.L. Tanimoto. "Pict: An Interactive Graphical Programming Environment," *Computer*, Nov. 1984, pp. 7-25.
[2]K.N. Lodding, "Iconic Interfacing," *IEEE Computer Graphics and Applications*, March 1983, pp. 11-2O.
[3]K. Potosnak, "Do Icons Make User Interfaces Easier to Use?" *IEEE Software*, May 1988, pp. 97-99.
[4]K. Matsumura and S. Tayama, "Visual Man-Machine Interface for Program Design and Production," *Proc. IEEE Workshop Visual Languages*, CS Press. Los Alamitos, Calif., 1986, pp. 71-80.
[5]M. Hirakawa, M. Tanaka, and T. Ichikawa, "An Iconic Programming System: Hi- Visual," *IEEE Trans. Software Eng.*, Oct. 1990, pp. 1,178-1,184.
[6]K. Kojima, Y. Matsuda, and S. Futatsugi, "LIVE: Integrating Visual and Textual Programming Paradigms," *Proc. IEEE Workshop Visual Languages*, CS Press, Los Alamitos, Calif., 1989, pp. 80-85.
[7]D.N. Smith, "Visual Programming in the Interface Construction Set," *Proc. IEEE Workshop Visual Languages*, CS Press, Los Alamitos, Calif., 1988, pp. 109-120.
[8]D.C. Halbert, "Programming by Example," Tech. Report TR OSD-T8402, Systems Development Dept., Xerox Office Systems Div., Palo Alto, Calif., Dec. 1984.
[9]D.L. Maulsby, I.H. Witten, and K.A. Kittlitz, "Metamouse: Specifying Graphical Procedures by Example," *Proc. ACM SIGGraph*, ACM, New York, 1989, pp. 127-136.
[10]W. Finzer and L. Gould, "Programming by Rehearsal," *Byte*, June 1984, pp. 187-210. ■