* * * DRAFT * * *

DDN SERVICES/HP3000

INVESTIGATION REPORT

HEWLETT PACKARD

INFORMATION NETWORKS DIVISION

Location Code: 66-7860

Project Number: 6651-1131

July 9, 1986

David St. John

PRODUCT IDENTIFICATION

SECTION 1

Name	HP/3000 DDN Services
Mnemonic	none
Project Number	6651-1131
Project Manager	Doug Heath (FTP) Bruce Templeton (Telnet) Peggy Garza (SMTP)
Project Engineers	David St. John (FTP) Mark Laubach (SMTP) Katy Jenkins (Telnet)
Product Manager	Dennis King
Product Assurance	Mike Mixon

PROBLEM STATEMENT

SECTION 2

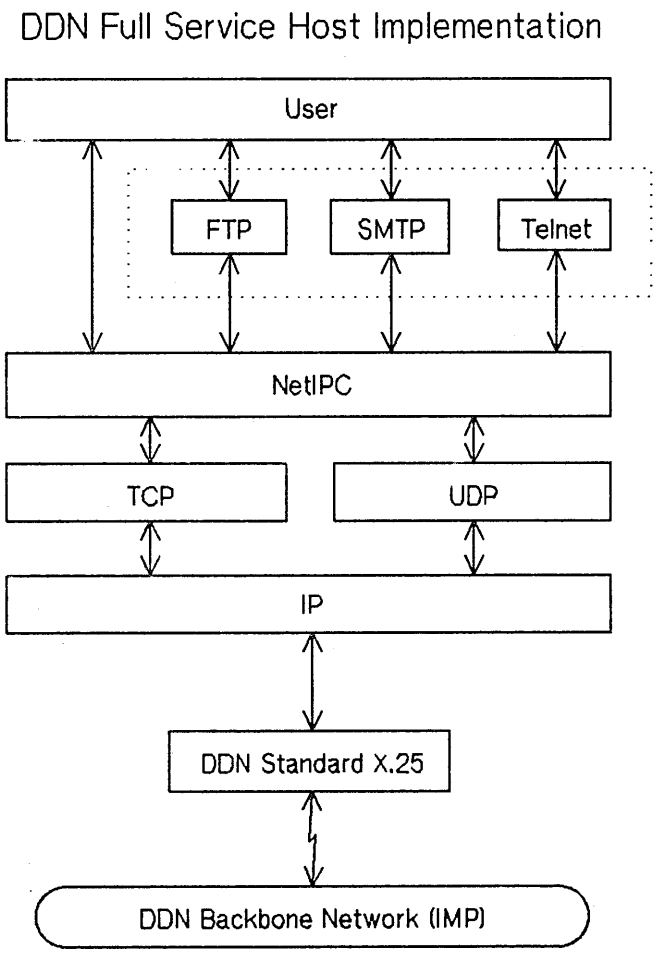
Hewlett-Packard has no official product available to its customers which allows them to communicate with a remote host via the DDN network. The current NS and DS products do not meet the DDN standards as issued by the Department of Defense. This document addresses the investigation of the effort required to provide HP3000 customers with DDN Services for file transfer and simple mail transfer. A product overview will be given in this section with a brief description of parts of that product which are not included in this report.

The DDN standards are designed to allow communications between heterogeneous systems. The standards are not nearly as flexible as those offered by NS. Since NS can be used on the DDN network, in some cases HP customers can achieve better services by using the NS product if it is available.

The investigation report is divided into the following sections:

2) Problem Statement
3) Marketing Analysis
4) Investigation Results
5) Telnet Investigation
6) Quality Perspective
7) Implementation Schedule
8) References

2.1 PRODUCT OVERVIEW



The above figure shows the conceptual layer model for full DDN services as now specified by the Department of Defense. The area in the dotted box, File Transfer Protocol (FTP), Simple Mail Transfer Protocol (SMTP), and Telnet (network virtual terminal protocol), are DDN Services which will be provided by this product.

2.2 SUB-SERVICES LAYERS

DDN Services are dependent upon the layers below being DDN compatible and certified as such. Projects are under way to make these meet those standards.

2.2.1 X.25 Link

This part of the product consists of levels 1, 2, and 3. Level 3 is being developed at Grenoble Networks Division. The X.25 product must meet the X.25 Host Interface Specification[6] and must be certified as such as outlined in DDN Host Interface Qualification Testing, Link and Network Layers[7] before the Services can begin DDN qualification testing.

2.2.2 Network Transport

This is an umbrella term which includes the following:

- Internet Protocol (IP)
- Transmission Control Protocol (TCP)
- User Datagram Protocol (UDP) [NOT required for this product and not scheduled for transport]
- NetIPC (Network InterProcess Communication)

First release of the network transport will not provide transparent access for DDN Services from nodes not directly connected to the DDN network (see section 5.7, DDN Compatibility for Netxport: Investigation Report,[12]). The release of DDN compatible services is dependent upon the modifications of NS/3000 Network Transport as outlined in DDN Compatibility for LAN/3000: Investigation Report[12] which should be read for more detail about implementation and testing requirements for these layers.

MARKETING ANALYSIS

SECTION 3

3.1 SUMMARY

The following Defense Data Network (DDN) marketing analysis addresses the following four questions:

1. Is there sufficient market need to warrant the development of a set of DDN services?

2. If there is sufficient demand, what does the market require of the product?

3. What is the cost benefit to the Division of developing the DDN services?

4. What is the lost business potential if we don't develop a set of DDN services?

The growth of the DDN over the next 3-4 years should be significant according to growth projections in the BBN Future Technologies Study. The results of this study indicate that the number of hosts on the DDN could grow to 20,000 from the present 400 by the end of FY1989. The most conservative estimate indicates that the growth could be an order of magnitude less or 2000.

The DDN protocol requirements are specified in various MIL standards. In the Request for Proposals (RFP's) that have come in, the Simple Mail Transfer Protocol (SMTP), File Transfer Protocol (FTP) and Telnet Protocol have all been required. Selling all three of the protocols together as a bundle addresses the needs of most customers and provides all of the functionality required.

The results of the parametric studies done for this analysis indicate that this product will be profitable for the Division. Three scenarios were considered based on the BBN study:

1. An "Optimistic" case - The number of DDN hosts will grow from 400 now to 20,000 by the end of FY1989 (US govt FY).

2. An intermediate case - The number of DDN hosts will grow from 400 now to 10,000 by the end of FY1989.

3. A "Worst" case - The number of DDN hosts will grow from 400 now to 2,000 by the end of FY1989.

Additional assumptions were:

1. 5% of the incremental DDN business will go to HP.

2. The current HP3000 installed base at the DoD is 200 systems and will remain at this level until we have a DDN product.

3. Due to the government procurement cycles and DDN market needs, only HP3000's will be sold to the government through FY1989, and not Spectrum.

Prices considered for each case were $6000, $8000, and $10,000 (Prices did not include HPDesk required for SMTP or any of the NS/3000 Network Services). For the lowest price and the "worst" case, the following was determined:

*IRR = 63%

*NPV = $167,000

The cost of **not doing** a DDN product was assessed by considering revenue lost on potential product sales to the government, and more importantly, revenue lost on new incremental system sales. Incremental system sales are assumed to be either HP3000 Series 4X or HP3000 Series 58 machines. Total system sales lost as a result of not doing a DDN product based on the "worst" case estimate is **80 systems over 3 years** or about**$16M**

Product revenue lost to the Division would be **$1.68M** over three years (This number is derived by taking the low product price of $6,000 and multiplying it times the sum of the installed base, 200 systems, and the new system sales, 80 systems. or $6,000*280 systems = $1.68M). This lost revenue estimate does not take into account additional lost revenue from the sales of ATP's, INP's etc.

3.2 DDN MARKET ANALYSIS

3.2.1 DDN Overview

DoD data communication requirements are expanding rapidly. The purpose of the DDN is to meet these requirements. Packet switching technology developed for the Advanced Research Projects Agency Network (ARPANET) enables the DDN to achieve this purpose with a high degree of economy and performance. Over the past decade, the ARPANET has provided a research and development environment for state-of-the-art techniques in data communications. The DDN is a direct beneficiary of the ARPANET accomplishments. The DDN is employing ARPANET technology and, in fact, is absorbing a major portion of the existing ARPANET, as well as other military networks that use the ARPANET technology.

The communications services that the ARPANET has been providing since the late sixties are significant because they enable computer systems from different vendors, with different operating systems to exchange data. The data can be files, programs, or electronic mail. This type of communication is known as heterogeneous host-to-host communication. The vehicle for providing this type of communication is a layered protocol architecture Vendor specific protocol architectures such as HP's AdvanceNet or IBM's SNA provide similar services for a particular vendors product line, or homogeneous communication.

The advantage offered to DDN subscribers by a protocol architecture that supports heterogeneous communication is significant; a very diverse group of users and their software systems can interoperate. Such interoperability ensures that critical DoD systems will be able to communicate with one another in the future.

Recently, after years of design, implementation, and testing by the Defense Advanced Research Projects Agency (DARPA), the ARPANET protocol hierarchy was enhanced. The enhancements broadened the scope of the architecture to include multiple interconnected networks. In extending the architecture to span network boundaries, no assumptions were made about the underlying communications technology of each individual network. This permits subscribers with networks using different technologies, such as local area networks, to interoperate with ARPANET subscribers. In 1983, DCA divided the ARPANET into two separate networks, the MILNET and the ARPANET, thereby acknowledging the changing nature of the military communications environment, and forming the unclassified segment of the DDN.

The DDN is using this protocol architecture for its DoD subscribers. The enhancements are embodied in the Transmission Control Protocol (TCP) and the Internet Protocol (IP), which are DoD standards. As DoD standards, these protocols form the basis for ongoing security R&D efforts which, over time, will be incorporated into the DDN without impacting subscribers.

The DDN protocol suite provides a set of interoperable subscriber services. In the research community, all subscribers have implemented the **complete** protocol suite; therefore, all ARPANET subscribers could communicate with all other ARPANET subscribers. Since the DDN subscriber community is vast, with an accompanying set of unique requirements, waiver procedures have been established to permit subscribers to utilize the DDN even though they may not have implemented all of the required DoD standard protocol connections at the time. However, it is incumbent upon all subscribers (and also to their benefit) to implement the **full DDN protocol suite in a timely manner**.

3.3 MARKET REQUIREMENTS

HP has a significant opportunity to take a leadership role in the DoD market place by working with the Defense Communications Agency (DCA) to offer a networking product for our computers which meets the requirements of the Defense Data Network (DDN) protocols. With such a product, HP will be able to pursue the large DoD network requirements being planned for the next 5 years. Without this capability, which in essence is a "lockout spec", HP will be unable to pursue DoD computer business which involves networking. It will allow HP to bid on new business opportunities as well as focus on the business of the HP3000 DoD installed base. In addition, HP would realize add-on business to DoD contractors who will also require DDN compatible networking capability to secure future DoD contracts.

3.3.1 Market Size

The results of a preliminary survey done in the field (conducted by the Rockville sales office and based on the Governments GSA figures) shows that there are approximately 8000 computers installed in DoD agencies (not all on the DDN).

Currently there are 400 host computers (all vendors) and 305 Ethernet type networks on the DDN (Source DDN Network Information Center). DCA projections indicate that the number of host systems on the DDN will exponentially grow to over 20,000 host systems by the end of FY89.

3.3.2 DDN Market Requirements and Needs

The Director of the DCA initiated a study in September 1981, to access the capabilities of the AUTOMATIC NETWORK II, and to evaluate a plan for an alternative that could be used instead. The purpose of the study was to describe a survivable, common-user datacommunications system. In April 1982, the DoD terminated AUTOMATIC NETWORK II, and directed that the DDN be implemented as the DoD common user data communications network. On the basis of this decision, guidance from the Secretary of Defense now states:

All DoD ADP systems and data networks requiring data communications services will be provided long-haul and area communications, interconnectivity, and the capability of interoperability by the DDN. All existing systems, systems being upgraded and expanded, and all new ADP systems or data networks will become DDN subscribers. All such systems must be registered in the DDN User Requirements Data Base (URDB). Once registered in the URDB, requests by a Service (e.g., Air Force or Navy) or Agency for an exception to this policy shall be made to the Deputy Under Secretary of Defense.

3.3.3 Market Timing

The first major indication of the DoD'S intention to make DDN a requirement for future business was the AAMUS RFP. Based on other RFP's, it is clear that all DoD agencies are becoming fully compliant with this directive. The U.S. Army has mandated that "*all Army agencies will ensure that their future ADP acquisitions are tailored to utilize the DDN as the pursuing data communications media*" and has put out guidelines as to how this requirement will be incorporated in future procurements. The Air Force has put DDN requirements into their computer requirements, and the group evaluating office automation in the Navy has said that DDN will be a requirement.

DDN Certification

Currently, the only hard date for DDN certification is January 1986 for Basic X.25. This means that all new hosts being added to the DDN and any existing host that will be moved must pass the current DDN certification tests conducted by DCA. This rule does not apply to existing hosts that will not be moved (after talking to the DDN PMO, it does not appear that the rule will be changed in the near future). Given that leased lines will no longer be in use after this date, the rule also implies that a number of service agencies will have to apply for waivers until the hosts are DDN compatible at layers 1 and 2. Also, DCA feels that certification at layers 1 and 2 is a must now, because any damage to the DDN will probably occur here and not at the higher layers, which is why certification at IP and TCP are not as clearly defined.

Current thinking within the DCA is that certification for IP and TCP should be under the direction and control of the services (i.e.,Air Force, Army and Navy etc). Since there are no dates for transport layer certification, it is difficult to anticipate market timing for the DDN transport. The only indication of timing for the transport is the new business anticipated in the forecast section. DCA's position is that X.25 certification is a must at this point, and that complying with the MIL-STD Specs for IP and TCP is a strong measure of our commitment to the DDN strategy. The final assumption is that in the future there will be a number of AMMUS type RFP's coming and that having the DDN protocols implemented will allow HP to be responsive in a timely way. As data presents itself, it will be incorporated.

3.4 PRODUCT DESCRIPTION

3.4.1 DDN Services and Protocols

(FTP) and Simple Mail Transfer Protocol (SMTP) are standard DDN application protocols. They support scroll mode terminal-to-host communication, file transfer service, and electronic mail service. The DCA recommends that each subscriber host implement TELNET, FTP, and SMTP.

In addition to the application protocol implementations, user interfaces to the application protocols must be provided. The user interface portions of these applications are not standardized, and the specification of their functionality is HP's responsibility.

3.4.1.1 Data Transport Services and Protocols

The ARPANET Transmission Control Protocol and its associated Internet Protocol are the standard DDN transport protocols. The TCP/IP protocols provide the reliable host-to-host peer level communication necessary to support the application protocols above.

3.4.1.1.1 Transmission Control Protocol. TCP provides a reliable datacommunications service for interprocess communication over the DDN and other TCP/IP networks connected to the DDN. It is connection oriented; that is, it maintains a connection, or virtual circuit, for each pair of communicating processes. TCP incorporates mechanisms to ensure the reliability of connections and to control the flow of data over the connections.

3.4.1.1.2 Internet Protocol. IP transmits and receives data across the DDN and networks connected to the DDN. Unlike TCP, it is connectionless; it treats each packet as an independent entity. Furthermore, it neither checks user's data for errors, nor performs flow control. Instead, its purpose is to provide a means for communication across multiple networks. To this end, it supports a global addressing system, and it accommodates differences in maximum packet sizes allowed by networks.

3.4.1.2 Network Access Protocols

The network access protocols define the interface between a host and the network . The DDN can be accessed by way of an X.25 interface. Concerning X.25, there are two types of service available:

- Basic X.25 Services: this is the default. This type of service allows communication between systems of the same type (e.g. HP3000 to HP3000), or using compatible higher level protocols.

- Standard X.25 Services: they are required for communication between systems of different types (e.g. HP with non-HP), and implies the use of DDN higher level protocols

GND is responsible for generating the Standard X.25 Services. In addition, the current X.25 product will be certified for use on the DDN with the Basic X.25 Services. This certification will provide the services (Navy,etc) with the ability to communicate HP3000 to HP3000 over the DDN. This complies with the DCA strategy regarding migration of subscribers onto the DDN and will offer an adequate solution until we are ready to bring our DDN product to the marketplace.

3.5 DDN STRATEGY

Networking is one of Hewlett-Packard's key strengths. Developing and offering a DDN product is consistent with our AdvanceNET strategy, and would establish HP as a leader in the DoD marketplace. The development of a DDN product will allow HP to penetrate the government marketplace and create closer business relationships with key R&D groups within the DoD, especially the DCA. In turn, HP could help guide future DoD plans on networking. Certification is vitally important to the services both from operational and budgetary perspectives. To this end, the Services (i.e., Navy, Air Force, Army etc.) would lobby in our behalf regarding future certification issues if they perceived that having a powerful DDN product would give them enhanced DDN capability. Our leadership role will also enhance our position as a supplier to defense contractors (Hughes, Lockheed, Northrup, etc.). On the other hand, without a timely DDN product, HP will be unable to pursue any DoD computer requirements which involve networking. This would restrict or eliminate our growth in this marketplace.

3.5.1 Current Product Strategy

The DoD requires that all DoD systems with a requirement for internetting conform to TCP/IP in order to ensure their interoperability. The subscriber may choose to attach temporarily to the DDN using DDN Basic X.25, and to exchange data between application level functions using suitable vendor supplied

end-to-end protocols. Guidance from the DCA indicates that new systems and systems that have been or will be moved will have to be certified for the Basic X.25 Services on the DDN by January 1986.

For DS and X.25,IND will certify the product at the link and network levels. This certification is currently scheduled to be completed during the first calendar quarter of 1986. This certification will allow system to system communication between HP3000's connected to the DDN.

3.5.2 Future Product Strategy

The base set of DDN protocols consists of the the physical layer, the link layer, the subnet, the internet, and the transport layers. At the link and subnet layers, IND will offer and support both the X.25 and IEEE 802.3 protocols for DDN. The X.25 protocol will be in accordance with the CCITT specs except where FIPS PUB 100 and the DDN X.25 Host Interface Specifications offer exceptions. Data obtained from the DDN Network Information Center indicates that there are currently 305 Ethernet type networks on the DDN. The likelihood that one of our HP3000's will be connected to one of these LAN's means that we will have to have the capability to communicate with other non-HP hosts. The ARP (Address Resolution Protocol), which will be discussed later, would simplify this. In addition, we can provide local area networking to those subscribers that desire this capability.

At the Internet layer, IND will implement a version of the IP protocol in compliance with MIL-STD-1777. For TCP, IND will implement the protocol in compliance with MIL-STD-1778.

3.5.2.1 Upper Layer Protocols

The DDN services are those protocols that reside above the Base protocols. At this level, current planning calls for us to implement the following protocols:

- Simple Mail Transfer Protocol (SMTP)
- File Transfer Protocol (FTP)
- TELNET (Virtual Terminal)

Since SMTP, FTP and TELNET are "checkoff" items on most RFP's, the best approach is to provide the minimum functionality that meets the intent of the specification.

With regard to SMTP, we will use HPDESKMANAGER as the user interface and plan at this time to sell the interface and the service separately. SMTP, FTP and TELNET will be implemented in accordance with MIL-STD 1781, 1780 and 1782 respectively.

3.6 BUSINESS POTENTIAL

ASSUMPTIONS

1. The installed base (IB) of HP3000 computers at the DoD is 200 systems. (Source: IND Product Marketing installed base database listings)

2. The number of hosts on the DDN will grow to 20,000 by the end of FY1989. This represents the most optimistic case. (Source: BBN Future Technologies Study)

3. The most conservative case for DDN growth is 2000 hosts on the DDN by the end of FY1989. (Source: BBN Future Technologies Study)

4. The most probable case for DDN growth is 10,000 hosts on the DDN by the end of FY1989. (Source: Data Communications Dec. 1985)

5. There are currently 400 hosts on the DDN. (Source: Data Communications Dec. 1985)

6. 5% of the incremental DDN business will go to HP. (Source: HP Federal Marketing Organization - FMO)

7. The trade discount is 26% (Source: HP FMO)

3.6.1 Pricing and ROI

A financial analysis for the DDN Services was conducted to determine the Return on Investment (ROI) and the Net Present Value (NPV). Three forecast scenarios were considered:

1. An "optimistic" case - The number of DDN hosts will grow from 400 now to 20,000 by the end of FY1989 (US governments FY).

2. A "worst" case - The number of DDN hosts will grow from 400 now to 2000 by the end of FY1989.

3. An intermediate case - The number of DDN hosts will grow from 400 now to 10,000 by the end of FY1989.

The model used for the analysis included a trade discount of 26% that is given to the government (Data was obtained from HP FMO). Financial data (Field Selling Cost, Allocated Overhead, Administrative, Overhead, Lab and Marketing rates) were obtained from IND Finance. Lab/Engineering level of effort were obtained from the IND Datacomm DDN lab team, and the Marketing estimates came from IND product marketing. The spread sheet for the "worst" case scenario is presented at the end of this analysis.

In addition, three prices were considered for the product, $6000, $8,000 and $10,000. These prices did not include the incremental cost that a customer would have by buying HPDESK as the user interface for SMTP, or the addition of NS/3000.

Using the above parameters, a set of ROI and NPV values were obtained. These values are summarized in table 2 below. Based on the results of this study, a DDN product would be profitable for even the most conservative scenario considered.

SCENARIO #1 - Low/"worst" case forecast

Using the assumptions given, the following represents the estimated total number of DDN products that HP could sell to the government over the next 3-4 years.

1. DoD HP3000 current installed base = 200 systems (Data obtained form IND Product Marketing System Installed Base listings)

2. DDN Growth = 2000 systems - 400 currently installed = 1600 new systems on the DDN.

3. New HP3000 system sales = .05*1600 systems = 80 new systems on the DDN (This assumes that 5% of the Government sales will go to HP)

4. Total sales potential = Installed base + New system sales = 200+80 or 280 systems on the DDN over the next 3-4 years

SCENARIO #2 - HIGH/"Optimistic" case

1. DoD HP3000 current installed base = 200 systems

2. DDN growth = 20000 - 400 = 19600 new systems on the DDN

3. New HP3000 system sales = 0.05*19600 = 980 over the next 3 years

4. Total sales potential = IB + NSS = 200 + 980 = 1180

SCENARIO #3 - INTERMEDIATE case

Although true DDN subscriber demand is probably close to the 20,000 hosts by the end of FY1989, the delays in integrating complex subscriber equipment into the network will probably place the real growth somewhere in the middle of the two previous scenarios (Source Data Communications/Dec. 1985). The basic assumption for this scenario then is that **the DDN will grow to 10,000 hosts by the end of FY1989.** This leads to the following projection:

1. DoD HP3000 current installed base = 200

2. DDN Growth = 10,000 - 400 = 9600 new systems on the DDN by the end of FY1989.

3. New system sales = 0.05*9600 = 480 system that can be sold to the end of FY1989

4. Total sales potential = IB + NSS = 200 + 480 = 680 systems on the DDN that will need a DDN Product.

FORECASTS AND ROI

Based on the above projections, the following tables show the forecasts and ROI values.

	1986					1987				1988				1989				1990	
(	HP FISCAL QUARTERS)																		
	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2	3	4	1	2
LOW FORECAST	0	0	0	0	0	0	2	7	12	19	28	33	40	68	71	64	53	47	42
HIGH FORECAST	0	0	0	0	0	0	10	30	50	80	120	142	173	290	305	275	225	200	180
INTERMEDIATE FORECAST	0	0	0	0	0	0	6	17	28	45	68	81	98	164	173	156	128	113	102

Table 1 - DDN Services Forecasts (Low, High and Intermediate cases)

		DDN FORECAST SCENARIOS		
		LOW/"WORST"	INTERMEDIATE	HIGH/"OPTIMISTIC"
P R O D U C T	$6000	IRR = 63% NPV = $0.27M	IRR = 143% NPV = $1.09M	IRR = 213% NPV = $2.16M
	$8000	IRR = 86% NPV = $0.46M	IRR = 176% NPV = $1 56M	IRR = 257% NPV = $2.98M
	$10000	IRR = 105% NPV = $0.65M	IRR = 205% NPV = $2.02M	IRR = 295% NPV = $3.8M

Table 2 - ROI & NPV As Functions of Volume (Forecast) and Price

3.6.2 Lost Business and Revenue to HP Without a DDN Product

The cost of **not doing** a DDN product was assessed by considering the lost revenue to the Division by not having DDN, and more importantly, the lost revenue to the Corporation on HP3000 system sales and peripherals. The new or incremental system sales are assumed to be either HP3000 Series 4X or HP3000 Series 58 machines. Using the conservative projections and forecasts, the result of not doing the DDN product will be **80 systems and peripherals not sold** to the Government over the next 3 years.

Revenue lost to the Division would be $1.68M over the next 3 years (This estimate was derived by taking the low product price $6,000 and multiplying it times the sum of the installed base, 200 systems, and the new system sales, 80 systems. Or $6,000*280 = $1.68M). This estimate does not account for additional revenue that will be lost due to the sale of additional ATP's, INP's etc. that could be sold.

3.7 CONCLUSIONS/RECOMMENDATIONS

Based on the marketing research done to date and the financial analysis conducted, it appears that there is sufficient market need and interest to warrant developing a set of DDN services. Also, by not developing a DDN product, we will be locking ourselves out of future government procurements that will require DDN compliance. In addition, not doing a DDN product will force many of our government customers to go to other vendors who are DDN compatible in order to comply with DCA DDN guidelines.

The DDN services will be a profitable product. Considering the most conservative financial estimates and assumptions leads to the conclusion that we will experience a ROI of 63%.

It is recommended that IND develop and market a set of DDN services for the HP3000 computer family.

At some time in the future, the DoD may want to migrate over to the Spectrum family of computers. Due to the timing of Spectrum, the present market need and the fact that the government is slow to change, the emphasis needs to be on the HP3000 In addition, due to the way budget authorization and expenditure are handled by the DoD, there is usually a one to two year time lag in the procurement process. Therefore, if the DoD is ordering for FY1988, they usually begin their planning during FY1986

or early 1987. This means that Spectrum related datacomm should be positioned when we are ready to sell Spectrum machines to them.

INVESTIGATION RESULTS

SECTION 4

This section details the changes that would be required to existing products as well as additions that would have to be made for DDN Services. An effort has been made in the investigation to integrate DDN Services with the current NS/3000 product as much as possible. This was considered desirable for several reasons.

- The release of DDN Services would be possible at an earlier date if existing data structures, procedures, and routines were used.
- Changes required to MPE should be minimized if those made for NS/3000 are used in a more generic manner.
- Integration of DDN Services within NS/3000 will require fewer overlapping system resources. This is especially true if NS and DDN services are both installed on one system and operating at the same time.
- Integration of DDN Services within NS/3000 will provide a uniform interface for the Network Manager. Making a network-independent interface should minimize the amount of training required for installing, configuring, and maintaining the DDN product.
- Note that we are not proposing that DDN Services be sold as a part of NS/3000 Services, but rather that some routines be separated from the current NS product as a separate product used by many network services (see Section 4.1.1).

The investigation results are divided into the following subsections:

4.1) NS/3000 Session Services changes for DDN Services
4.2) Host Names
4.3) File Transfer Protocol (FTP)
4.4) Simple Mail Transfer Protocol (SMTP)
4.5) Miscellaneous issues
4.6) Summary of Dependencies
4.7) Risks and Contingencies

4.1 NS/3000 SESSION SERVICES CHANGES

NS/3000 Session Services will be used as a control agent for DDN Services in a manner similar to the current NS product. For SMTP no servers or initiators will be used since these processes will be a part of HPDESKMANAGER running on the system. The service must be allowed via the NSCONTROL command before SMTP can transmit or receive a message over the network. FTP initiator processes will be created via a ":RUN" command issued by a user or via the CREATEPROCESS intrinsic called in a user program after it has been started in the NSCONTROL command. Therefore FTP will not rely upon a pool of local servers. A pool of FTP remote servers will be made available by DSDAD to handle RemCnctReq messages. These FTP servers will operate as children of DSDAD until the FTP user process passes enough information for the server to request session creation from the operating system and adopt itself under that session.

4.1.1 NS Product Restructuring

We propose that certain elements of the current NS product be separated into a unique product to be used by this and future services. The following modules of NS/3000 are essential for DDN services, although others may be required by those products who will also be using an NS core product.

1) Module 10 - DSDAD control process for network servers.

2) Module 12 - ASCX1SEG and ASCX2SEG for NSCONTROL executor.

3) Module 13 - ASBUFSEG for buffer management.

4) Module 15 - DSUTIL for global tables, port-related routines, error logging, session startup and termination, version checking, etc.

5) ASCAT catalog.

4.1.2 Nscontrol Changes

The NSCONTROL command will be altered to control DDN services as well as the current services.

- START[=services] function will have to be changed to accept FTP, FTPL, SMTSEND, and SMTPRECV character strings as valid services. START with no options could start all services purchased by the customer. This necessitates a mechanism for determining what services have been purchased. This is being investigated by the NS/3000 CPE group and will be completed before release of DDN Services.

- STOP[=services] will have to be altered as the START function above.

- SERVER function will accept FTP as a valid server name.

- LOG function will NOT be used by FTP or SMTP. The network manager can use the function to log events of DSDAD, but in line with other NS services it was decided not to implement this. Tracing will be released to customers for both services. The local user can use a TRACE ON/OFF command. Enabling and disabling trace for a server process will require a new NSCONTROL TRACE = ON/OFF,<PIN> command similar to the DEBUG parameter.

- STATUS function will be supported for DDN services. Changes must be made to include these services, but not the users, in the status report similar to the current NS NFT.

- VERSION function should be altered to report DDN modules versions and/or NS modules.

4.1.3 DSDAD Changes

DSDAD will have modifications to handle two new servers, FTP and SMTP. Every attempt will be made to make these new servers as similar as possible to the current servers, DSSERVER and NFT.

- DSDAD must be able to handle the NSCONTROL changes above in the NscontrolReq message from CXNSCONTROL executor.

- The following decimal port numbers are reserved for FTP and SMTP. These correspond to the current SAP addresses.

  1) Port 20 - FTP Data Connection.
  2) Port 21 - FTP Control Connection.
  3) Port 25 - SMTP Connection.

NOTE

DSDAD should only create a service initiation socket for the FTP Control Connection. The FTP Data Connection is only done by the FTP server process. The SMTP Connection will be owned by the SMTP server although it will check for the presence of the its name in the Port Dictionary before initiating.

- Two new pseudo-service initiation ports will have to be stored in the Port Dictionary. Although the FTP and SMTP service requesters will not be sending a ServiceReq to their local "L" port, a call to DICTFIND will be done to ascertain if users have been allowed to use them in the NSCONTROL START command.

4.2 HOST NAMES

Although the two protocols under investigation and Telnet are the only ones that are required for DDN certification, one other feature is deemed necessary for their proper functionality. A mechanism must be provided to translate node, or host, names as provided by the user into an address to be used in making a connection to that host. Since there is no required or recommended protocol for name servers, Official ARPA-Internet Protocols, RFC944[13], it has been decided to use a static host Network Directory until a later release. Name server protocols were examined for inclusion in Phase I release. Domain Name protocol is experimental and has not been scheduled for implementation by the DDN-PMO. The elective Hostname protocol was examined more closely and was considered as a viable method for interactive addition of entries into the Network Directory. It was decided not to use it for the following reasons:

1) There was no clear evidence that the DDN market would like to have the facility.
2) The protocol is only elective and may be replaced before the release of the product.
3) The Domain Name protocol is in use in much of the ARPA community and may be the preferred one in the near future.
4) The extra expenditure in resources may be wasteful for such a possibly short-term solution.
5) Testing of the protocol requires a connection to the SRI-NIC, but a connection cannot be obtained unless the lower layers are qualified for inclusion in the DDN network.

4.2.1 Network Directory

The Phase II NS transport project will use a Network Directory for keeping information about nodes which can be accessed on the network. A separate project in IND is responsible for implementation of the Network Directory. The directory as it now stands is not fully defined. However, since it is a requirement for Phase II transport and is to be used in the same way we wish to use it, we will plan on using it to access host names and IP addresses entered by the system Network Manager. In this way access to the network can be controlled by system management. The progress of the Network Directory project will be monitored in order to insure provisions will be made for our needs.

4.2.2 DDN and NS Host Names

There will be a conflict between the node naming conventions as used by NS and those which are used by the DDN network. The NS naming conventions (see AdvanceNet Naming [16]) are:

env_name.domain_name.org_name

where all three are alpha [alphanumeric | "-" | "_"]... (max. 16 characters each)

DDN host names are specified in RFC952 [10] as:

alpha [alphanumeric | "." | "-"]... (max. 64 characters total)

Currently if an NS name is not fully qualified, defaults are used for the unspecified domain_name and/or org_name; For DDN no defaults should be added to the name. DDN Services will accommodate NS as well as DDN host names. This will allow DDN services to be used between "NS nodes" on a LAN, for example.

Currently the maximum length of a registered DDN name is 23 characters [10]. We recommend that Network Directory allow a maximum of 63 characters. Currently much of the ARPA community has

adopted the Domain Name Server Protocol which has a maximum length of 63. This will provide more space to accommodate this expansion without difficulty in the future.

NetIPC (in IPCDEST and IPCLOOKUP) should provide a special option which identifies the name space (e.g., NS or DDN). DDN will try this "DDN" option first. If that search is unsuccessful, the DDN services will recall the procedure without this option. Thereby NetIPC can use AS'VALID'NODE procedure to add NS defaults to the host name in order to determine if it is an NS-style name. This will save the user from the burden of fully qualifying an NS name, which is in keeping with the practice for the NS product. This option does not have to be limited to only the two node name styles under discussion. It is assumed that the change will have the flexibility to include future networks that may be developed at HP.

Network Manager will also have to be changed to allow an HP host to have two different node names: an NS-style name and a DDN-style name. Having an "NS node name" will allow a node to be accessed via the NS services without having to modify NS services to allow DDN-style node names. This access could be over a LAN or even over DDN (provided the remote "NS node name" is configured in the Network Directory). At the same time the node can be accessed by other nodes on the DDN via its "DDN node name." If a node has both a DDN and an NS name, both should be recognized as the node's "local name." This will require changes to transport and to Network Manager products.

4.2.3 NETIPC Changes

A few alterations must be made in the current NetIPC to allow certain requirements for DDN services. These changes may require a few minor changes to Transport, although none are foreseen which will require any thing not already proposed for DDN TCP/IP support.

- The above mentioned change to allow minimal checking of DDN node names.
- NetIPC will also have to make an alteration to IPCCONNECT to allow the FTP server process to specify the source port (#20) for the data connection. It is currently possible to include source port information in the IPCCREATE procedure and this can be used as an example for modifying the code.
- The source IP address and port of the FTP command connection will have to be made accessible to the server process. This port is the default destination port for the data connection.
- Fully specified passive open will have to allow the inclusion of the source IP address and port from which it is willing to receive a connection. This will insure that the correct data connection will be made in the FTP program.
- The FTP server process will have to open a data connection based upon the IP address and port of the remote. Currently NetIPC will only make connections based upon the remote host name. In most cases this will be unknown to the FTP server and, even if it were, it would have to be verified in the local directory or by some other means. We propose that connections be allowed using IP address and port as well as names.

4.3 FILE TRANSFER PROTOCOL

4.3.1 Introduction

File Transfer Protocol (FTP) is the DDN standard for transferring files between computer systems. The objective is to provide sharing of files by transferring data reliably and efficiently while shielding the user from variations in file structure and storage in different hosts. The protocol also allows other file manipulations, such as renaming and deletion of remote files.

4.3.2 Commands

Military Standard, FTP [1] specifies that every DDN-compatible host must provide a minimum subset of the complete command set. Other commands are optional. The following subsections are divided into three: the minimum implementation subset, those that are desired for increased functionality, and commands that could be implemented if resources are available. Note that "command" in this section does not necessarily mean user command Although they may coincide (see section 4.3.3), these are internal commands sent from a user FTP process to an FTP server.

4.3.2.1 Minimum Implementation

The following subset is necessary for any host to be certified with DDN. A brief explanation will be given with each command.

USER - User name string to identify the user on the remote host.
QUIT - Log off user on remote host.
PORT - Specifies the data connection port and host address to be used if the default port on the initiating host is not. This will be necessary if the initiator is neither the producer nor the consumer or if the default port is unavailable.
TYPE - Data representation type (i.e. ASCII, binary). This command has several parameters which will be discussed in section 4.3.2.4. Default values are ASCII, non-printable.
STRU - Structure of specified file (e.g. file, record). The options for this command will be discussed in section 4.3.2.5 below. Default value is File.
MODE - Data transmission mode. Different options are explained in section 4.3.2.6. Default value is Stream.
RETR - Retrieves a file and transfers it via the data connection.
STOR - Causes the server to accept a file via the data connection and to store it on the host.
NOOP - This command does not change any previously entered commands or parameters, but requires the remote server to respond.

4.3.2.2 Further Recommended Commands

This subset of commands could also be provided for FTP users on the HP3000. There were two criteria used for this list: 1) the functionality of the product would be increased substantially by their inclusion, and 2) the implementation of these commands would require a minimum of added engineer resources.

PASS - User password necessary for logon. This can be specified in the USER command, but it would be minor to respond to a reply code 331 (User name okay, need password) in case the user forgets to use the password in the logon string.
ACCT - Account name necessary for logon. This can also be stated with the USER command, but allows recovery in case reply code 332 (Need account for login) is returned from the server.

APPE - This commands the server to accept data from the data connection and to store the file. If the file exists, append the data; otherwise create the file and store the data.
RNFR - Rename the specified file. This command must be followed by the RNTO command.
RNTO - Rename the file specified in the preceding RNFR command to the file name specified with this command.
ABOR - Aborts the data transfer and closes the data connection.
DELE - Delete (purge) the specified file from the host.
NLST - Specified file directory listing will be transferred from the server to the user via the data connection.
ALLO - Allocate a specified number of bytes for storage of file.

4.3.2.3 Remaining Commands

These commands finish the list specified in the FTP Mil-Std. They are considered low-priority items for either a later release or for first release if resources are sufficient. These are provided in no particular order and should be considered separately if it is decided to implement any of them.

HELP - Returns the command set available to the remote user.
PASV - This command informs a server that it should listen on a data port rather than initiate action. The reply contains the host and port on which the server will listen. This will be useful for three host transfers in which the initiator, producer, and consumer are operating on different systems.
REST - The argument field contains a server marker at which file transfer is to be restarted. It must be followed by an appropriate command (RETR, STOR, APPE) to resume data transfer.
REIN - Re-initializes the command connection (i.e., logoffs the user, flushes I/O after data transfer is complete and resets to the default settings).
CWD - Allows the user to work within a different directory.
SITE - This command provides access to local services necessary for file transfer that are not provided by the protocol standards.
STAT - The user can ascertain the status of the remote host.

4.3.2.4 Data Represetation Types

The required command, TYPE, takes two parameters which relay the type of data which will be transferred over the data connection. The second parameter is referred to as the Format parameter. Only the Non-Print format parameter will be available for first release. Military Standard, FTP[1] specifies that ASCII type must be supported by all implementations and recommends that IMAGE (i.e., binary) be supported; both will be supported upon first release. Other parameters are not considered important enough to be included at this time.

4.3.2.5 Data Representation Structures

Three file structures are defined by the standard:

1) File - implies no internal structure of file
2) Record - data divided into sequential records
3) Page - file divided into independently indexed pages

File and Record must be implemented for first release. Page structure is not planned at this time.

4.3.2.6 Data Representation Modes

Three data modes are specified in the FTP standard:

1) Stream - stream of bytes with control codes for EOF and EOR
2) Blocked - data divided into blocks with headers
3) Compressed - data compression when appropriate

Stream mode is the default and will be the only one available upon first release since both file and record structures can be supported using this mode. Blocked data and compression may be implemented on later releases.

4.3.3 User Interface

The user process of FTP will be a program which the user may run. This will alleviate the need for any changes to the MPE Command Interpreter. The user will have a help facility available since not all of the FTP commands used between hosts will be available and many of them will be implemented using other commands (e.g. RNFR and RNTO will be replaced by the one command, RENAME). The user command set will be based 1) upon common MPE file manipulation commands (e.g. PURGE for deleting a file) and then 2) other FTP implementations of other vendors for continuity and ease of use. The server FTP process will also be a program file. The file system and various components of NS (see section 4.1) will be used making release of FTP independent of a particular release of MPE.

4.3.4 Programmatic Interface

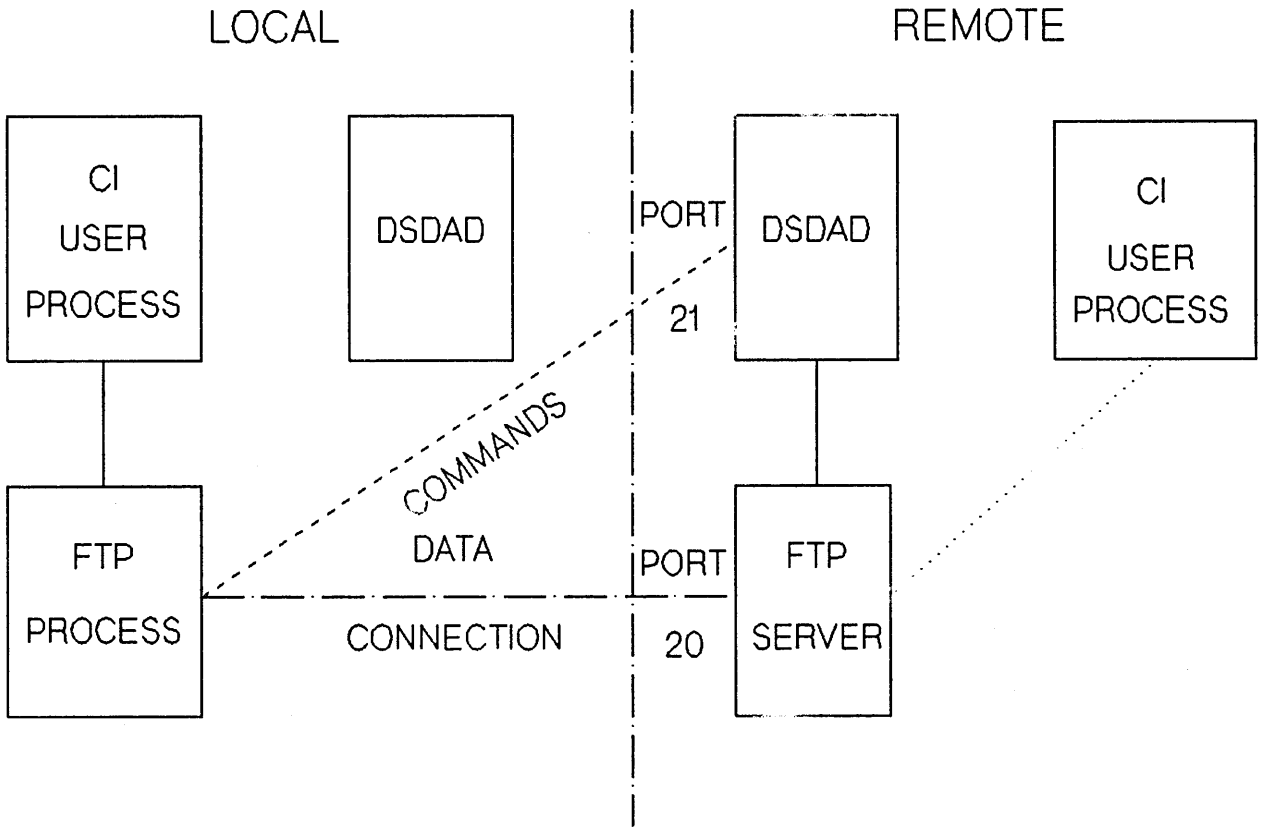
The Mil-Std states, "FTP, although usable directly by a user at a terminal, is designed mainly for use by programs." Thus programmatic access to the FTP program will be via the CREATEPROCESS MPE intrinsic. This mechanism is already available and it was decided to use it rather than create a new intrinsic used only for DDN. The user documentation should have a full explanation of this feature, as well as examples.

4.3.5 Testing and Certification

There is a draft, Defense Data Network Host Interface, Qualification Testing: Higher-Level Protocols [15], which discusses test procedures for certification with the DDN. There will be rigorous testing of FTP not only between HP3000's but also with at least one non-HP system (e.g. VAX-11 with DDN Services from Berkeley UNIX 4.2BSD), and an HP9000 series 300 or 500. Different file types and sizes will be transferred in both directions as well as error conditions created to test detection and recovery. We shall track the finalization of FTP qualification specifications and will consider it a necessity to comply with them.

File Transfer Protocol: Remote Driver Specification, [17] describes test scenarios and explains the mechanism of the remote test driver to be used for qualification. This will be included in the test package.

4.3.6 FTP Internal Structure



FTP Service Internal Structure

The above figure is a brief sketch of the current plan for implementation of FTP. It is not meant to be definitive and may be changed during a later phase of the development. It is only intented to give the reader an idea of the structures necessary for FTP. The above diagram shows both sides in a connection. Either side could be a non-HP3000, which would not affect the design. This was only done to simplify the diagram.

1) The user runs the FTP program and MPE creates the FTP user process.
2) The FTP process checks the presence of Port FTPL in the Port Dictionary and if present makes a connection to port 21 of the requested host on which the remote DSDAD process is listening.
3) DSDAD creates a FTP server process and gives the connection to it.
4) When the FTP server receives remote user and account information from the FTP user process, the server will have MPE create a session using this information. The FTP server will then adopt itself into the remote session.
5) Upon a data transfer request, the FTP server will open a data connection to the FTP user process, the data will be transferred, and the connection will be closed.
6) When the FTP user closes the command connection, the remote FTP server will adopt itself back to DSDAD and delete the remote session.

7) The FTP user process will close the connection to the remote host and terminate itself, returning the user to the CI command level.

4.3.7 FTP Implementation

The only version of FTP for the HP3000 known by this team is that written by BBN and modified by the El Paso, Texas HP office for use at White Sands Missle Range. It is currently unsupported. TCP/IP versions are also available. The porting of this code for release as an HP product was rejected for several reasons.

1) The code is written in SPL which is not portable to Spectrum.
2) There is only a user FTP program, the server process code was never written.
3) The code was written for the Series III and would have to be changed for MPE-V including such things as port procedures.
4) It is known that field engineers in the El Paso, Texas office attempted to convert the code for MPE-V and had major problems, although they never did respond to requests for a copy of their conversion.
5) The lower level interface was to their own version of TCP/IP and would require major revision to be useable with those being developed at IND.
6) The user interface is not well developed e.g., the user must use RNFR and RNTO in order to rename a remote file.

The other implementation that was examined closely was the Berkeley 4.2BSD Unix. This has a more presentable user interface and is a released, widely used, and accepted product. This will be used as a guide for the HP product. It has been decided that it would be better to develop the product by the lab rather than convert the Berkeley code for the following reasons:

1) The code is meant to operate in the Unix environment and relies greatly upon that factor.
2) The lower-level interface is to Berkeley sockets which would have to be altered to work with the HP TCP/IP DDN product.
3) The code is written in C and it is not known at this time if the recently released C compiler for the 3000 is stable enough for product development.

We feel as though the complete implementation of FTP by IND will require less effort than porting this version. It will serve as the chief model of non-HP implementation against which we will measure our product.

A third party supplier was considered for FTP. It was determined that the in-house implementation of FTP would not require a great resource in engineering time considering the advantages of having this part of the product match the HP quality of software and the control that HP would have in post-release product support.

4.4 SIMPLE MAIL TRANSFER PROTOCOL

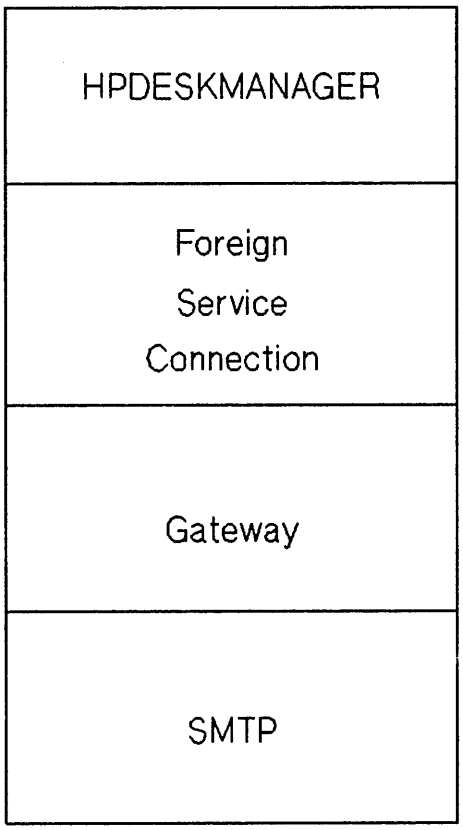
4.4.1 Introduction

SMTP is DDN standard for transferring mail reliably and efficiently over the DDN network between hosts. SMTP is also widely used on non-DDN systems, such as DEC VAX 4.2/4.3 BSD, Sun Microsystems, and Apollo workstations.

This project consists of two distinct parts, the Simple Mail Transfer Protocol [SMTP] server system and the HPDESK <-> DDN Gateway. The SMTP portion is responsible for moving mail on and off the HP 3000 in an environment of heterogeneously manufactured systems. The Gateway portion is responsible for the proper translation of message formats, addresses, and semantics between HPDESK and the single messaging standard used by the "foreign" system.

HPDESKMANAGER will serve as the user interface for this service, providing the HP customer with network independent transparency. The figure below shows the relationship between HPDESKMANAGER and SMTP.

HPDESKMANAGER/SMTP Interface



The Foreign Service Connection [FSC] is currently a part of HPDESKMANAGER which provides an interface between itself and non-HPDESKMANAGER electronic mail systems. The Gateway, which will be addressed in greater depth, provides translation between the foreign service formats and those of SMTP as specified in Military Standard, SMTP [2].

4.4.2 SMTP Commands

4.4.2.1 Minimum Implementation

The minimum command set as specified in Military Standard, SMTP [2] must be available on first release.

HELO - Identifies the sender-SMTP to the receiver-SMTP.
MAIL - Initiates a mail transaction to one or more mailboxes.
RCPT - Identifies an individual or multiple recipient of the message.
DATA - Signals the start of the transfer of the mail message.
RSET - Specifies that the current transaction is to be aborted.
NOOP - Has no action other than to force the remote host to send an OK reply.
QUIT - Informs the receiver process that the connection is closing.

4.4.2.2 Further Recommended Commands

This subset of commands should also be provided for the SMTP system on the HP 3000. These commands augment the MIL-STD set and move towards a better compliance with the RFC821 spec used by the ARPANET (see Internet Mail Protocols, [11]). The benefit from having these commands will be a more friendly user environment.

VRFY - Asks the receiver to confirm that the argument identifies a user.
HELP - Causes the receiver to send helpful information to the sender.

4.4.2.3 Non-Implemented Commands

The following SMTP commands will not be implemented in any version of the SMTP subsystem for reasons of security or the lack of an available procedural interface with HPDESK.

EXPN - Asks the receiver to expand the mailing list and return the membership of that list.
TURN - Asks the receiver to turn roles and become the sender-SMTP.
SAML - Mail and deliver to one or more terminals.

4.4.3 Command Reply Codes

The following SMTP reply codes are generated by the receiver as a response to the commands issued by the sender-SMTP process. Most command reply codes will be implemented in accordance with Internet Mail Protocols, [11].

500 Syntax error, command unrecognized [This may include errors such as command line too long]
501 Syntax error in parameters or arguments
502 Command not implemented
503 Bad sequence of commands
504 Command parameter not implemented
211 System status, or system help reply
214 Help message [Information on how to use the receiver or the meaning of a particular non-standard command; this reply is useful only to the human user]
220 <domain> Service ready
221 <domain> Service closing transmission channel
421 <domain> Service not available, closing transmission channel [This may be a reply to any command if the service knows it must shut down]
250 Requested mail action okay, completed
251 User not local; will forward to <forward-path>
450 Requested mail action not taken: mailbox unavailable [E.g., mailbox busy]
550 Requested action not taken: mailbox unavailable [E.g., mailbox not found, no access]
451 Requested action aborted: error in processing
551 User not local; please try <forward-path>
452 Requested action not taken: insufficient system storage
552 Requested mail action aborted: exceeded storage allocation
553 Requested action not taken: mailbox name not allowed [E.g., mailbox syntax incorrect]
354 Start mail input; end with <CRLF>.<CRLF>
554 Transaction failed

4.4.4 Gateway

The gateway will perform the conversions necessary between the SMTP protocol and HPDESKMANAGER. This implementation will require no changes for HPDESKMANAGER. The following items must be addressed:

- HPDESK addresses must be converted into SMTP addresses as specified in Internet Mail Protocols, [11] and related, more recent RFC documents. The Network Directory will be used to resolve host name to address as recommended earlier. Addresses from the DDN network will be translated into a format that the HPDESK user would expect.

- A version of the product is currently in place and in use in an HP internal HPDESK <-> HPUNIX electronic mail gateway product developed and supported by Corporate Engineering. Incorporation of the existing gateway into a first release of the product will require minimal effort.

4.4.5 HPDESK/SMTP Interface

There are a few limitations on the part of both SMTP and HPDESKMANAGER. Some of these can be easily handled but others can not be. Those features or limitations which will impact the customer will have to be well documented.

- HPDESK allows a message to contain several parts which is not allowed in SMTP. All of these parts will be concatenated together into one text message. Only ASCII data are allowed, so any parts which are not will be returned. Binary files can be transferred using FTP service. (Note here that 8-bit "ASCII" will be supported.)

- SMTP provides no provision for marking an item as URGENT; setting this flag in HPDESK will not be passed to the recipient.

- No provision is made for acknowledgment of a message beyond the fact that it was delivered; therefore the ACKNOWLEDGE command will only be supported up to level 2 for messages sent over the DDN.

- On incoming mail all header information as specified in RFC822 will be preserved in the MESSAGE HEADER part of the HPDESK message.

- Full distribution lists will be translated where possible.

- The gateway must handle any address limitations of HPDESK. SMTP guarantees return addresses for incoming and outgoing mail. Mail will be returned if HPDESK can not handle the address of the remote sender.

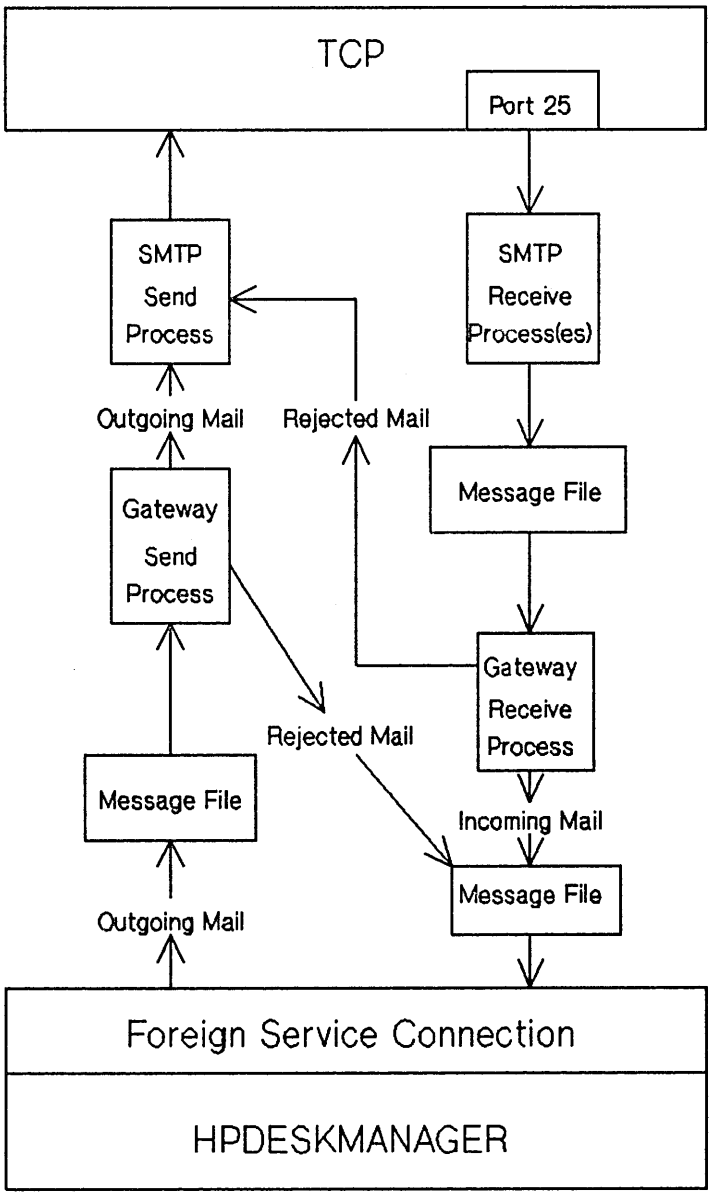
- Foreign mailing address host strings (domain specs) are limited to 33 characters maximum.

- HPDESK does not allow messages marked as PRIVATE to be sent out an FSC port.

- The primary Subject field of an HPDESK message is conveyed properly across the gateway. However, due to a functional limitation of HPDESK, the Subject field associated with a part, package, or copied file is lost when mail leaves HPDESK via the FSC interface.

4.4.6 SMTP Processes

The SMTP part of the figure will consist of two processes, one for sending and one for receiving. The receiving process will accept messages from the network and pass them on to the gateway for conversion. The receiving process will be a family member of HPDESK, but will only have access to the network if the Network Manager has allowed the service via the NSCONTROL command, as outlined in section 4.1.



SMTP Internal Structure

The above figure represents the basic process and data flow layout of the HP 3000 SMTP/Gateway

system. This figure is intended to give the reader the basic idea of the process steps necessary to gateway mail between the external world and the HPDESK system.

Outgoing Mail

1) The HPDESK system places outgoing messages (via the FSC interface) into group HPMAIL.HPOFFICE and posts a message record on the output IPC file.
2) The Gateway Send process picks up the message and performs the necessary text and address format conversions and places the converted message in the outbound queue HQTOUX for pickup by the SMTP Send process. Messages that cannot be processed are returned to the HPDESK originator via the HQTODESK queue.
3) The SMTP Send process picks up the mail and attempts to initiate a connection directly to the recipient host machine or nearest message relay machine for the current message and recipients. Successfully transmitted messages are deleted from the queue. Messages which cannot be delivered due to some irrecoverable error or due to a host being unavailable for a long period of time (configurable) will be returned to the HPDESK originator via the HQTODESK queue. As per ARPANET/DDN conventions (standards), the sender-SMTP process, when connecting to a remote host will only attempt connection to TCP port #25.

Incoming Mail

1) The SMTP Receive father process listens for connection requests on incoming TCP port #25. A son process is spawned to perform the actual work.
2) The SMTP Receive process places mail on the HQFRMUX queue and posts a message record on the input IPC file.
3) The Gateway Receive process picks up the message and performs the necessary text and address format conversions and places the converted message in the inbound queue HQTODESK for pickup by the HPDESK FSCAREF process. Messages that cannot be processed are returned to the external originator via the HQTOUX queue.
4) The HPDESK FSCAREF process accepts inbound mail and transfers it into the HPDESK system.

4.4.6.1 DSDAD Changes

The port dictionary will be updated by DSDAD as outlined in section 4.1.3. The send process will be responsible for transmitting the mail message to the network. It and the receive process will be family members of HPDESK. They will both be responsible for making their own connections to the network.

4.4.6.2 HPDESK Coordination

The SMTP/Gateway mail server system will be initiated/terminated in coordination with the methods already in place for HPDESK. An actual study of the design criteria for this item are not in place yet; however, the general goal is that the SMTP/Gateway system should only be running when HPDESK is running.

4.4.7 Testing and Certification

A large part of the implementation has been in place and used heavily within Hewlett-Packard as part of the internal HPDESK <-> HP Unix electronic message gateway project developed by the R&D Information Resources Group of Corporate Engineering. The experience provided by this system is providing rigorous testing of the gateway components.

The SMTP server will require additional testing over that provided above. Testing will include communications certification between the HP3000 and the standard SMTP packages provided on the

following systems: 4.2BSD VAX, 4.3BSD VAX, HP9000 S200/S300/S500 (when available with their DDN support), Sun II workstations, Sun III workstations (if available), Symbolix systems, and Apollo workstations.

As with FTP the DDN Host Interface Qualification Testing, Higher-Level Protocols, [15] progress will be monitored in order to guarantee that the product meets the qualification testing when it has been finalized. As of this time there is no specification for a SMTP remote driver [15] as there is for FTP [17].

4.4.8 SMTP Ownership

SMTP will be based upon the current internal HPDESK <-> HP Unix electronic message gateway project developed by the R&D Information Resources Group of Corporate Engineering. That group will be responsible for making any conversions to meet the SMTP standards. After coding and module testing the code will be given to the Network Resources Lab of Information Networks Division for completion and support as outlined in Product Life Cycle, e.g., reliability testing, documentation, IMS, etc.

The future direction of the SMTP product in relation to a general strategy for electronic mail transport will be addressed in a forthcoming investigation from the Network Resources Lab.

4.5 MISCELLANEOUS

This part outlines miscellaneous results of the investigation which should be covered in this report:

1) Porting issues
2) Resources required for implementation
3) Performance estimates

4.5.1 Porting Issues

The product team is considering it a requirement that this implementation will take a minimum of effort to port. This includes all code written by this team whether that be for any part of this product or for tests. Some of this code will be dependent upon the file system, operating system, mail system, or other system-specific interfaces. We shall aim to isolate that code as much as possible so only certain modules must be altered for system-dependent code. We shall be principally concerned with native mode Spectrum, including HPE and HPUNIX, although the latter is planning on using DDN Services being developed at FSD for the 9000 series. Using these guidelines and the rules below, porting may also be easier for the 1000 series. Contact will be maintained with those engineers who are porting the current NS product as well as with the HPE, HPUX, and 1000 teams for issues that may arise later. Among the rules that will be observed are:

1) The code will be written in a language that will be fully supported on Spectrum. SPL will definitely NOT be used. C, Pascal, and Modcal are currently being considered. Whatever the language, HP3000 specific code will be avoided as much as possible (e.g., ANSI Pascal, Pascal I/O procedures will be used, etc.).
2) Split stack mode will not be used.
3) MPE intrinsics will be used rather than using HP3000 machine instructions to accomplish the same thing (e.g., DMOVEIN instead of MFDS). This does not override rule 1 above. Language provided procedures will be used if they can accomplish the same end.
4) HP3000 machine instruction set will in principle be avoided.
5) Low-level memory addressing and system table addressing will not be used.
6) PDISABLE and PENABLE will not be included.

4.5.2 Required Resources

This section does not address the implementation schedule which is covered in the next section, but rather the system resources necessary to release this product in a timely manner.

An HP3000 will be required for initial testing of the modules in IND. SMTP is being developed at Corporate Engineering.

The main resources for the reliability testing of the DDN combined product should be based in the Network Test Center. The first phase of testing is planned to include two HP3000s interconnected on a LAN with an HP9000 series with DDN services software being developed in Colorado. The second phase of testing will include DDN-compatible X.25 DDN-compatible TCP/IP. We should have the same systems as used in the first phase as well as a non-HP machine e.g., a VAX. As well as using a LAN we will also want to include point-to-point connections in order to utilize the X.25 software.

A connection to the DDN network will be an absolute requirement before certification can begin. It may be required at an earlier date since both X.25 and TCP/IP must be certified in advance.

4.5.3 Performance Issues

It will be a goal of this product team that performance will at least be as good as a Berkeley 4.2 implementation running on a system comparable to an HP3000.

4.5.3.1 SMTP Performance

The SMTP and gateway processes will little influence the amount of time to transfer a mail message to a user via the DDN network. The time will not be noticed by the user since HPDESKMANAGER transport is a batch job operating in the background. The transport of a message over DDN should be no slower than over other comparable networks.

4.5.3.2 FTP Performance

FTP, as an interactive process, will be more noticeable to the user. Once again it is the stated goal of the team that performance will at least match a comparable system. Several transfers of the file RFC810 were made from the SRI-NIC to a VAX11/370 at HPLABS. The transfers were made at random times during a period of 6 days. The average throughput was 0.36 K/sec. to transfer 14659 bytes of data. The throughput should be comparable to that of the current NS NFT interchange mode.

4.6 SUMMARY OF DEPENDENCIES

Since the dependencies are scattered throughout the report, this is a summary of those with references to where the discussion can be found.

4.6.1 Non-DDN Services

DDN X.25: Standard X.25 must be DDN compatible and certified (2.2.1). This is being developed at Grenoble Networks Division (prog. mgr: Claude Cornet). [Available 10/1/86; Certified by 2/16/87].

DDN Transport: Network Transport must meet the Mil-Std for TCP and IP (2.2.2). This is being developed as part of Phase II of NS/3000 Transport (proj. mgr.: Kevin Faulkner). Node name and related changes needed by 8/1/86; dual names changes by 8/1/86 (4.2.3). ARP needed by 8/1/86 (4.2.2). Changes required for NetIPC (4.2.4). [Completely available for testing 10/1/86; certified by 2/16/87.]

NS/3000 Session Services: Separation of part of NS and changes in order to control other networks, including NSCONTROL and DSDAD (4.1). This is being done by NS/3000 services CPE project (proj. lead: Charles Knouse). [Needed by 7/1/86].

Network Directory: Host table repository (4.2.1). This is being developed by Network Resources Lab (proj. mgr.: Clint Cuzzo). [Needed by 8/1/86].

Network Manager: If DDN is to be available on a LAN, changes must be made for allowing ARP protocol (4.2.2). [Needed by 8/1/86] DDN and NS double names (4.2.3). [Needed by 8/1/86]. This is being developed by Network Resources Lab (proj. mgr.: Clint Cuzzo).

4.6.2 DDN Services

TELNET: This service will be developed and supported in IND. Proj. mgr.: Bruce Templeton.

SMTP: After coding this service will be given to the NRL. Proj. mgr.: Peggy Garza.

FTP: This service will be developed and supported in IND. Proj. mgr.: Doug Heath.

4.6.3 Resources

Systems: At least one standalone HP3000. At least one HP9000 with DDN-compatible services software from CNO. A non-HP system, such as a VAX with DDN Services software. Two 3000 connected to the two above systems (4. 5. 2).

Network Connection: A connection to the DDN network for final testing phase. (4. 5. 2)

4.7 RISKS AND CONTINGENCIES

A word should be said about the risks that may arise during the development of DDN-compatible software and the contingencies which will be considered in case any of the dependencies are either behind schedule or cancelled.

The only required protocols as of this moment are the three outlined in section 2. The DDN-PMO could at any moment add one or more "required" protocols to that list. We have made contacts at the Defense Communication Agency in Washington, D.C. and shall depend upon them for advance warning of any future requirements. It is highly unlikely that a new protocol will be declared required without allowing vendors the time to implement the new protocol. If a new requirement is put forth, an investigation will have to be done. If the current product is released in a timely manner as outlined here, any new protocol should not impact Phase I.

Qualification of DDN Services has been mentioned in this report. The requirements are currently still in draft. The progress of the qualification document will be followed closely. The basic purpose of qualification is to ascertain if the software meets the standards for those services. By following the standards, using advance copies of the qualification tests, and consulting the DCA about any fine details, we will be able to guarantee that the services will meet all qualification requirements.

Section 4. 6. 1 contained a summary of the dependencies of the services covered by this report. In this part we wish to discuss contingencies that could be used if any one of these are not ready in a timely manner for testing or release.

DDN X. 25: The testing phase could continue using the current LAN product. We feel that full-length reliability testing must be done with X. 25 for release of DDN Services. The schedule will slip for this phase of testing if X. 25 is not available. In case the X. 25 product is not DDN-compatible, this product can not be released. Certification of the services is dependent upon the previous certification of X. 25; if it is not certified, then the certification schedule will have to be changed.

DDN Transport: Reliability testing can continue without a DDN-compatible transport layer, but the reliability schedule will slip since it is based on having the modifications. As with X. 25, full reliability testing must be done with DDN-compatible transport before release of this product. The lack of dual name changes can also be accommodated by setting limits on the node names during the testing phase. If

the TCP/IP layer is not certified with the DDN, the services can not be certified and certification would have to be delayed.

NS/3000 Session Services: If the NS changes are not available by the time that testing begins, modifications will have to made to adjust for their absence. It could take upwards of one engineer month to write a substitute for testing. If it is not available by the time reliability testing phase begins, the reliability schedule will have to slide until it is. We are considering that reliability testing is not valid until all parts of the total product are considered stable.

Network Directory: We can adjust for the absence of a Network Directory Configuration interface by using a separate static host table on the system as used in other implementations.

Network Manager: If dual node names are not present for reliability testing, an alternative will be used to solve the problem. If it is not ready for final release, the best solution at this moment seems to use quotation marks around a DDN name in order to stop validity check of the name.

TELNET INVESTIGATION RESULTS

SECTION 5

This section presents the Telnet investigation results. It follows the overall format of the previous section (which presented investigation results for DDN services in general, followed by specific discussions of FTP and SMTP). Since many of the general comments for DDN Services apply equally well to Telnet, they will not be repeated here. This section will focus on specific issues relating to Telnet. Any global DDN Services issues mentioned previously (for example, host naming conventions) should be considered to apply to Telnet as well, unless explicitly stated otherwise within this section.

5.1 NS/3000 SESSION SERVICES CHANGES

Telnet will use NS/3000 Session Services as a control agent in a manner very similar to FTP. TELNET local processes will be initiated via a ":RUN" command issued by a user. Thus TELNET will not rely upon a pool of local servers. A pool of TELNET remote servers will be made available by DSDAD to handle RemCnctReq messages. These TELNET servers will operate as children of DSDAD until the TELNET user process passes enough information for the server to request session creation from the operating system and adopt itself under that session.

5.1.1 NS Product Restructuring

Telnet will use the NS/3000 modules provided in the NS/3000 core product. TELNET's module requirements are the same as those listed for DDN services in Subsection 4.1.1.

5.1.2 Nscontrol Changes

The following changes need to be added to the NSCONTROL command for Telnet.

- START[=services] function will have to be changed to accept TELNET, and TELNETL character strings as valid services.
- STOP[=services] will have to be altered as the START function above.
- SERVER function will accept TELNET as a valid server name.
- LOG function will NOT be used by TELNET.
- VERSION function should include TELNET modules in its report of DDN modules.

5.1.3 DSDAD Changes

DSDAD will have modifications to handle the new server TELNET. This server should look similar to the current servers, DSSERVER and NFT. This server should also be similar to the server developed for FTP.

- DSDAD must be able to handle the NSCONTROL changes above in the NscontrolReq message from CXNSCONTROL executor.
- The following decimal port number is reserved for TELNET. This corresponds to the current SAP address.

  Port 23 - TELNET Connection.

NOTE

DSDAD should create a service initiation socket for the TELNET Connection.

- A new pseudo-service initiation port will have to be stored in the Port Dictionary. Although TELNET will not be sending a ServiceReq to its local "L" port, a call to DICTFIND will be done to ascertain if users have been allowed to use it in the NSCONTROL START command.

5.1.4 DSUTIL Changes

TELNET may require changes to DSUTIL (in DSBREAK2) to notify the TELNET server if the user has entered a subsystem break character. These changes would be similar to the way in which a subsystem break character is handled for NFT currently.

5.2 HOST NAMES

TELNET will follow the node naming conventions which are established for DDN Services. See Subsection 4.2 for discussion of this issue.

5.2.1 NETIPC Changes

TELNET will require the following changes in NetIPC. These changes may require a few minor changes to Transport, although none are foreseen which will require anything not already proposed for DDN TCP/IP support.

- The changes discussed in Subsection 4.2.3 to allow minimal checking of DDN node names.
- TELNET will NOT require the special connection options required by FTP (Subsection 4.2.4). TELNET will always make connections based on remote host name.
- TELNET will require explicit control of the TCP push flag through the NetIPC interface.
- TELNET will require the ability to send and receive TCP urgent data.

5.3 TELNET PROTOCOL

5.3.1 Introduction

TELNET is a DDN standard with the primary goal of allowing a standard method of interfacing terminal devices and terminal-oriented processes to each other. It provides a fairly general eight-bit byte oriented communications facility.

TELNET is built upon the concept of a Network Virtual Terminal (NVT). All hosts map their local device characteristics and conventions to appear as a standard Network Virtual Terminal to the network. Hosts may negotiate options to alter the conventions used over a connection between two hosts. All hosts desiring to communicate via TELNET are required to support the characteristics defined by the NVT. Hosts are not required to implement options, although MIL-STD-1782 [3] recommends the implementation of six options described in its appendices.

The MIL-STD remarks that the TELNET protocol is also envisioned for use in terminal to terminal or process to process communication. It is neither desirable nor practical to provide this capability on the 3000. This TELNET investigation is intended for terminal to process communication only. The TELNET implementation will be composed of two logical halves, a user side and a server side.

5.3.2 Telnet Commands

A TELNET connection is a TCP connection used to transmit data interspersed with TELNET commands. TELNET commands are distinguished from user data by the presence of a special "escape" byte immediately preceding the command bytes. If the user data to be transmitted contains the escape byte, two escape bytes are sent to indicate this fact. TELNET commands are interpreted and acted upon as they occur in the data stream. These TELNET commands are the only form of communication between the local and remote TELNET processes - there is no special "control connection" between the two processes.

The TELNET commands defined in MIL-STD-1782 are listed below. These are internal commands sent between a user TELNET process and a remote TELNET server. These commands may be sent autonomously by a TELNET process or may be sent in direct response to a user request. Unless noted otherwise, these commands may flow in either direction (user to server or server to user).

NOTE

These commands have the indicated meaning only when immediately preceded by the TELNET command escape character IAC.

5.3.2.1 General Commands

5.3.2.1.1 Minimum Implementation.

NOP - No operation.

GA - The TELNET GO Ahead signal.

IAC - Indicates a user data byte with decimal value = 255.

DM - Part of the TELNET "Synch" signal. Used to clear the data path to the other TELNET process. Requires TCP Urgent notification.

Comments

The TELNET GA command was intended to help a user's local host operate a physically half duplex terminal such as an IBM 2741. A TELNET server process is supposed to send a TELNET GA command whenever the application process suspends waiting for terminal user input. The GA signal informs the TELNET user process that it is safe to "turn the line around" and accept data from the terminal user. (Premature line turnaround would block further output to the terminal until the user entered data. This blocked output may contain the prompt the user is waiting for before entering data.) Options exist to suppress transmissions of GAs for implementations which do not require them. Nevertheless, the default condition, unless explicitly negotiated otherwise, requires the server to send GAs. Sending of GAs is not required in the user to server direction.

In the real world, however, things don't appear to work quite this way. Examination of source code from several TELNET implementations and discussions with people familiar with various TELNET installations (including ones using IBM 2741 terminals) reveal that many servers simply don't bother to send GAs and also don't bother to negotiate the fact that they aren't sending them. In light of this, it appears that the most prudent approach to take regarding the implementation of the GA command is that the 3000 TELNET server should send a GA at the appropriate time (unless explicitly negotiated otherwise), however the 3000 TELNET user process should not depend upon receiving GAs from the remote server. This allows the 3000 to meet the specification outlined in the MIL-STD, yet also allows it to talk to most of the existing TELNET implementations. Receiving GAs are not a problem for TELNET user processes which don't use them - they are simply treated as a No-operation.

5.3.2.2 Option Negotiation Commands

5.3.2.2.1 Minimum Implementation.

SE - Indicates end of subnegotiation parameters.

SB - Indicates beginning of option subnegotiation parameters.

WILL - Indicates a desire to begin performing the specified option.

WONT - Indicates a refusal to perform a specified option.

DO - Indicates a request that the other TELNET process begin performing the specified option.

DONT - Indicates a demand that the other TELNET process stop performing a specified option.

Comments

A host is not required to initiate any option negotiations, however any negotiations it does initiate must follow the rules for option negotiation outlined in the MIL-STD. A receiving host is not required to accept any option negotiations, however it must indicate its acceptance (or refusal) to the initiating host, following the rules outlined in the MIL-STD.

5.3.2.3 User Control Function Commands

5.3.2.3.1 Minimum Implementation.

BRK - Indicates that the terminal user struck the "Break" key.

IP - The TELNET user "Interrupt Process" function.

EC - The TELNET user "Erase Character" function.

EL - The TELNET user "Erase Line" function.

5.3.2.3.2 Further Recommended Commands.

AYT - The TELNET user "Are You There" function.

5.3.2.3.3 Commands Not Implemented.

AO - The TELNET user "Abort Output" function.

Comments

These are commands normally sent from the TELNET user process to the remote TELNET server in response to a request from the terminal user. The MIL-STD gives a general description of the intended use and outcome of each of these functions, however, the requirements are far from exact. To quote from the MIL-STD:

"... that is, a system which does not provide the function to local users need not provide it to network users and may treat the standard representation for the function as a No-operation. On the other hand, a system which does provide the function to a local users is obliged to provide the same function to a network user who transmits the standard representation for the function."

The HP3000 has direct analogs to the TELNET IP, EC, and EL commands. A local terminal user on the 3000 invokes these functions by striking the [BREAK], [CONTROL]H or [BACKSPACE], and [CONTROL]X keys, respectively. The TELNET BRK command is simply an indication that the [BREAK] key was struck, so this command would map to the same functionality as the IP command for the 3000. Thus, the TELNET commands IP, EC, EL, and BRK should be implemented for first release.

The HP3000 has no standard, direct analogs to the TELNET AO and AYT commands (although some subsystems may provide such functionality in an indirect, subsystem-specific manner). Thus, according to the paragraph from the MIL-STD quoted above, a remote TELNET server running on the 3000 is not required to do any special processing for these commands. A 3000 TELNET server could choose to implement these commands based on the functionality gained and implementation difficulty.

The TELNET AYT command is basically a user-friendly perk and would be relatively easy to implement. The TELNET AO command, although very nice for the user, would be difficult to implement on the 3000, and therefore is not recommended for first release.

Although these user control commands are generally sent in the user to server direction, nothing in the TELNET specification prohibits them from being sent from server to user. A 3000 TELNET user process receiving these commands should probably treat them as a No-operation.

5.3.3 Telnet Options

The following options are described in the appendices to MIL-STD-1782. Although these options are not mandatory, it is recommended that they be implemented for first release, as they are listed as requirements in most RFPs (Requests For Proposals) we have received. No other TELNET options are considered important enough to be implemented at this time.

5.3.3.1 Binary Option

This option allows two hosts to negotiate the sending of eight bit binary data over the TELNET connection (default is seven bit ASCII).

Comments

The TELNET binary option specification only defines the data passing over the TELNET connection. It does not define the data interface between the TELNET process and the terminal user or application which is using it. Although a binary or "raw mode" interface is often assumed, the TELNET option specification simply states that the implementer of the TELNET binary option should consider issues of binary data transmission to and from both a process and a terminal.

On the HP3000, reading and writing binary data to and from a process is easy to implement. Writing binary data to a terminal is also straightforward. However, doing a true binary read from a terminal is difficult, since the TELNET user process does not know the length of data to be read.

The following solution to the problem of binary reads from a terminal is proposed. Rather than doing true binary reads from a terminal, the TELNET user process shall read eight bit data from the terminal using limited editing capabilities. The exact definition of this editing mode will be given in the External Specifications; however one example is that hitting (RETURN) will terminate a read.

Since the TELNET protocol is primarily intended for use by a real person sitting at a terminal, this proposal should accommodate many of the expected uses of binary mode (such as applications which interpret the eighth bit as alternate character sets or graphics). This proposal will, however, exclude such uses as hooking up a paper tape reader or cartridge tape to a terminal port. The capabilities and limitations of this mode will have to be well documented.

This proposal will also pass the binary option certification tests currently listed in DDN Host Interface Qualification Testing, Higher Level Protocols [15], since these tests do not require binary terminal input from the implementation under test (IUT).

5.3.3.2 Echo Option

This option allows two hosts to negotiate echoing of data characters over the network. Default for TELNET is that characters are echoed by the local terminal host.

Comments

This option is useful as it allows a remote TELNET server to control character echoing when reading passwords, etc. Consideration must be given to that fact that HP3000 applications may not request echo disabling until after writing a password prompt to the user. If the TELNET remote echoing option is enabled only when a server application requests echo disabling, network delays may cause a TELNET option request for remote echoing to reach the terminal user host after the terminal user has started to type in a password. There are several ways to work this problem other than always doing remote echoing (which is lousy for performance). It is simply stated here as an issue to be addressed.

5.3.3.3 Suppress Go Ahead Option

This option allows two hosts to negotiate eliminating the requirement of sending GAs following transmitted data.

Comments

Although use of the TELNET GA signal provides a simple half-duplex protocol and read trigger capability, enabling the Suppress Go Ahead option does not require a full-duplex physical terminal interface complete with typeahead. Enabling the option simply indicates that a TELNET process should not send or expect to receive GAs at the end of user data. Since this investigation has indicated that GAs are not in common use anyway (see subsection 5.1.2.1), use of this option just explicitly states this fact.

5.3.3.4 Status Option

This option allows two hosts to exchange information about options currently enabled, since redundant option negotiation is not allowed under the TELNET option negotiation rules.

5.3.3.5 Timing Mark Option

This option allows two hosts a simple mechanism for synchronization.

Comments

One possible use of this option is for critical output verification, that is, completion of a server's FWRITE is delayed until the data is sent to the user's terminal. (It is unlikely, though that this use will be provided at first release.) No explicit user/application interface command to access this option is foreseen. The implementation of this option would simply handle requests for timing marks as specified when they are received over the network.

5.3.3.6 Extended Options List Option

This option simply defines a method of negotiating TELNET options when greater than 256 options exist. Supporting the Extended Options List Option does not imply support of any of these new options; it just indicates that a host understands the syntax for negotiating (accepting or refusing) these new options. Currently, approximately 30 TELNET options exist.

5.3.4 User Interface

5.3.4.1 User System

The TELNET user process will be a program which the user can run This will alleviate the need for any changes to the MPE command interpreter. This also means that the user will have to complete all transactions with the remote system and close the connection before being able to exit the program and enter commands to the local system. There will be no capability to switch between local and remote modes as exists in NS/3000. The benefits of this approach (independence from MPE, simpler design) are considered to outweigh the decrease in user friendliness of the product at this time.

The TELNET user program will have two input modes, command mode and data mode. Command mode will be used to open and close connections, and to enter special TELNET commands. Once a connection has been established, the user will enter data mode in which input/output is passed transparently to and from the remote host. The user may reinvoke command mode by entering a special character.

Since the TELNET specification does not provide a reliable mechanism for implementing read triggers, the TELNET user process will have to simulate a full-duplex interface by posting idle reads on the physical terminal. This approach will cause slightly different operational characteristics than those of a terminal running local applications on a 3000. These differences are not anticipated to be serious; nevertheless, a prototype user interface should be developed in the early phases of the project to assess their impact. User documentation should also explain these differences.

An ability to redirect input and output may be useful for testing. This capability does not need to be offered to customers unless someone sees a real need for it.

5.3.4.2 Server System

A pseudo terminal driver will be provided to capture application I/O requests intended for the TELNET session. This pseudo driver will communicate with a TELNET server process created by DSDAD when a connection request was received.

Block mode applications (including those using V/3000) will NOT be supported. This is because the TELNET protocol is not rich enough to exchange the control information necessary to perform terminal block mode I/O.

In theory, running over a TELNET connection should be transparent to supported applications since the I/O requests are intercepted at the I/O system level. In practice, the manner in which an application does I/O can greatly affect its performance over a network. Operations such as terminal status requests and timed reads may not work reliably. Additionally, the terminal user host system is not necessarily a 3000 and idiosyncrasies of its terminal I/O system may alter the way in which an application is perceived.

The External Specifications will contain a list of supported I/O request types.

5.3.4.3 The TELNET Network Virtual Terminal

MIL-STD-1782 specifies that the TELNET Network Virtual Terminal is a seven bit ASCII device. The eighth bit of user data is not guaranteed to pass through the TELNET connection unchanged. In order to send and receive eight bit data, the binary option must be enabled. The terminal user will be given a command to request the TELNET user process to negotiate the binary option. Applications may request the TELNET server process to begin negotiation through the use of standard FCONTROLS.

In addition to the 128 standard ASCII characters, the MIL-STD specifies that the Network Virtual Terminal be able to generate the following TELNET user control functions.

Synch (DM)
Break (BRK)
Interrupt Process (IP)
Abort Output (AO)
Are You There (AYT)
Erase Character (EC)
Erase Line (EL)

The 3000 TELNET user process will offer commands to allow the terminal user to request that the TELNET command for these user control functions be sent over the connection. Note that although a TELNET command will be sent, the remote TELNET server may not necessarily implement all of these functions.

The 3000 TELNET server process will not have any interface to allow applications to request that the corresponding TELNET command be sent since this is not their normal direction of use.

5.3.5 Null Terminal Capability

The implementers of FTP have indicated the possibility of requiring a null terminal capability from TELNET. A null terminal is managed by the TELNET pseudo driver and is used as a session device for programmatic logon. If this capability is required, the exact details should be worked out by the FTP and TELNET implementers.

5.3.6 Testing and Certification

5.3.6.1 DDN Qualification

There is a draft, Defense Data Network Host Interface, Qualification Testing: Higher-Level Protocols [15], which discusses test procedures for certification with the DDN. We shall track the finalization of TELNET qualification specifications and will consider it a necessity to comply with them.

TELNET Protocol: Remote Driver Specification, [18] describes test scenarios and explains the mechanism of the remote test driver to be used for qualification. This will be included in the test package.

5.3.6.2 Test Configurations

We currently plan to test and support the following configurations.

HP3000 to a VAX via X.25
HP3000 to HP3000 via X.25
HP3000 to HP3000 via 802.3
HP3000 to HP9000 Series 300 via 802.3

These configurations will be tested and supported in both directions (HP3000 as user node, HP3000 as server node). Other configurations which seem appropriate may be added as time permits.

A list of tested applications and configurations will be maintained.

5.3.6.3 Multiple Hops

It seems reasonable to expect that customers may want to use multiple hops, for example to communicate from a system on a LAN to a remote system which is connected via X.25 to another system on the LAN. No major problems inherent to this type of configuration are foreseen, however it will require significant testing. Multiple hops are seen as a high want for the product but not a must and are not required for first release. The multiple hop configurations which will be tested and supported will be specified in the Test Plan.

5.3.7 TELNET Implementation

The two implementations of TELNET which were examined by this team were 1) the HP3000 Series III code developed by Bolt, Baranek, and Newman Inc. and modified by the El Paso, Texas, HP office for use at White Sands Missile Range and 2) the Berkeley 4.2BSD Unix implementation. Neither implementation was considered feasible for porting for many of the same reasons discussed in the section under FTP (subsection 4.3.7). The Berkeley implementation will serve as a model of non-HP implementation against which to measure our product.

Another useful resource for the types of commands offered by a TELNET terminal user interface is the TAC User's Guide developed by BBN [19]. The TAC is a Terminal Access Controller used to connect terminals to the DDN.

5.4 MISCELLANEOUS

5.4.1 Porting Issues

Implementation of a pseudo driver will require TELNET to violate many of the implementation rules mentioned in subsection 4.5.1 (such as no SPL, no PDISABLE, PENABLE). In general, TELNET will try to minimize and localize the use of 3000 dependent code.

5.4.2 Required Resources

A standalone HP3000 will be required for initial testing of the TELNET modules in IND. The resources required for integration and reliability testing of the combined DDN product are discussed in subsection 4.5.2.

5.4.3 Performance Issues

The performance of TELNET running on the HP3000 should be comparable to that of using the Virtual Terminal Service of NS/3000. A direct comparison with a Berkeley 4.2 BSD implementation is difficult to make, since the physical terminal interface characteristics of the two systems are very different. In general, the TELNET performance should not differ radically from that of other implementations.

5.4.3.1 Performance Considerations for IPCSENDs

On the server side, TELNET will do an IPCSEND per user write request. The possibility of improving performance by concatenating multiple user write requests into a single IPCSEND exists; this may or may not be done for first release.

On the user side, TELNET will normally do an IPCSEND at the end of each line of user input (signified by the user typing carriage return or a defined alternate end of record character). We realize that there may be certain applications in which the user prefers that characters be sent to the remote side as soon as possible despite the overhead incurred. For these situations, we will offer a single character mode which must be explicitly enabled by the user. The default mode of operation will be line at a time, which should offer performance similar to that of NS/3000 VT.

As for incoming packets, TELNET has no control over how the other system sends data. UNIX systems, in general, are infamous for sending one-byte (of user data) packets, so performance expectations when connected to these systems will be less than when connected to another HP3000.

5.4.4 Project Priorities

Schedule is the highest priority for this project, followed by functionality, followed by performance.

QUALITY PERSPECTIVE

SECTION 6

This section first describes each of the global quality goals in descending priority order. These goals are prioritized so that if any conflicts between these goals arise, the conflicts can then be resolved in favor of the higher priority goal. Each objective has a minimum level of quality that is acceptable. The priority order is also used to achieve a level beyond this minimum. Then, in this section, there is a list of a few development guidelines and a brief discussion of some testing issues and objectives. The final part of this section is a brief list of a few general issues.

6.1 GLOBAL QUALITY GOALS

The definitions for the global quality goals are more or less general definitions and not, in most cases, project specific.

6.1.1 Reliability

This product will be reliable and it will function exactly to its specifications. Before release the product will pass 120 hours of reliability testing and will not have any known critical or serious bugs. A range will be determined for the number of normal or low bugs at time of release. Also, no system failures should occur because of this product.

6.1.2 Supportability

All documentation for this product will be in final form and ready to ship to customers and field personnel by MR. This includes the customer manuals, field training, and SE training. All internal documentation will be done at this time as well (IMS, code commentary, updated ES). A plan for the support of the product after MR must be in place which will include lab and marketing resources, further training development plans, further enhancement plans, etc. The field and factory support organizations will be trained in the product so it can be supported once it is released. Also, the product must be easy to install.

The division goal of MTTR (mean time to repair) bugs will be followed. These goals are 30 days to repair critical bugs and 60 days to repair serious bugs. Also the division goals of classification (30 days for medium bugs and 60 days for low bugs) will be followed.

6.1.3 Functionality

The goal for the functionality of this product is to insure that the entire feature set as described, and as intended, in the ES will be implemented in this product.

6.1.4 Performance

The desired performance for the product will be set early in the lifecycle. It will be measured, tested and evaluated before release.

6.1.5 Usability

This product, including the manuals, will be consistent and predictable. The error messages will be straightforward, useful and consistent. They will be documented in the manuals. This documentation will include information on diagnostics, i.e. what to do in case a particular error occurs.

6.2 DEVELOPMENT GUIDELINE

The following is a list of a few objectives for maintainability of the code.

1) The code shall be supportable by a non-author. Thus, the IMS and any other documentation necessary to support the code must be completed.

2) The code will be modular, include sufficient documentation and will be written in a clear and understandable style. All procedures (unless extremely simple) must have headers with the following information: function, input parameters, required conditions, algorithm, output parameters, and side effects.

6.3 TESTING

The following is a list of only a few of the major objectives. It is by no means a complete list.

1) Testing activities will be planned so that required resources and sufficient time can be committed.

2) An effort will be made to automate as many of the tests as possible.

3) We will insure that 85% of the executable code for each module will be tested.

4) We will track defects by making sure everything gets on STARS. Then we will be able to compare across projects and have a foundation on which to base our post-release reliability and to judge the proper time for release. We will start tracking the defects when the code is turned over for testing. We should also have someone (non-engineer) to enter the data into STARS.

5) Before the product can undergo system testing (i.e. stress, configuraton, subsystem, performance, etc.), it must pass 24 hours of reliability testing, so that we test on a stable base.

6.4 GENERAL ISSUES

1) We should try to write the IMS's much earlier than we have for some similar projects in the past, because they can help the new engineers become more familiar with the code. It might even be easier, and less of an inconvenience, to write them during the coding phase.

2) We will need a good configuration management (or version control) plan.

3) We need to develop the technology of test package design and the tools to keep the test packages updated as new defects are found. In this way we can build our regression test base efficiently.

4) We need to be more customer oriented in our test planning. This has been a major weakness in the testing effort for past projects: we were not customer-oriented enough in our test designing. This can be improved by visiting customers or having them donate copies of their applications for our test suites. Our customers would benefit from by knowing that their applications had already been tested, and we would benefit by having real representative cases to use here in the factory.

IMPLEMENTATION SCHEDULE

SECTION 7

The code size estimate, including comments, for the parts of the product covered in this report are:

FTP	-12000 lines
SMTP	-13000 lines
Telnet	-12000 lines

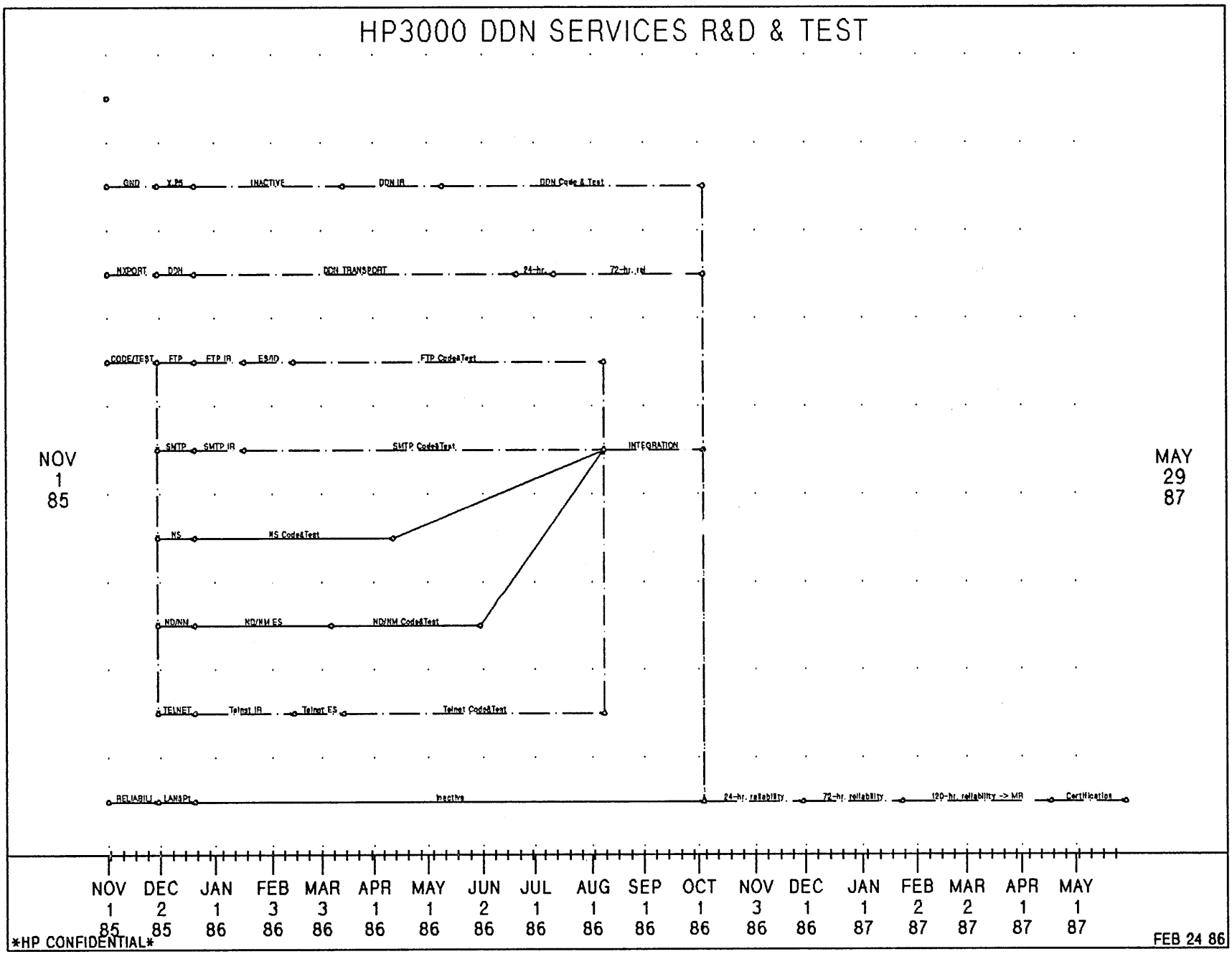
The following chart is the projected schedule for the release and certification of DDN Services based upon the availability of dependencies. Since the investigation of Telnet has not been completed, the date of its availability is only projected to meet the integration phase of all the components of the other DDN Services.

System resources have been discussed in 4.5.2. The following engineer resources should also be scheduled:
1 FTP Engineer and Organizer
1 SMTP Engineer for coding (Corporate Engineering)
1 engineer March 1986-March 1987 (IND)
1 Telnet Engineer
1 QA engineer
1 part-time NTC technician (beginning at reliability testing phase)

Major Testing Milestones	Start Date
Integration, except TCP/IP and X.25	8/1/86
24 hour reliability with TCP/IP, X.25	10/1/86
72 hour reliability, Alpha testing	12/1/86
120 hour reliability, Beta testing	1/26/87
Product MR	4/17/87
Certification testing	4/20/87

NOTE

The following schedule includes DDN-compatible X.25, but the investigation of the product has not been started. The schedule may have to be altered when the effort is better understood.


HP3000 DDN SERVICES R&D & TEST
GND
X.25
INACTIVE
DDN IR
DDN Code & Test
NXPORT
DDN
DDN TRANSPORT
24-hr.
72-hr. rel
CODE/TEST
FTP
FTP IR
ES/ID
FTP Code&Test
SMTP
SMTP IR
SMTP Code&Test
INTEGRATION
NS
NS Code&Test
ND/NM
ND/NM ES
ND/NM Code&Test
TELNET
Telnet IR
Telnet ES
Telnet Code&Test
RELIABILI
LAN&Pt
Inactive
24-hr. reliability
72-hr. reliability
120-hr. reliability -> MR
Certification
NOV 1 85
MAY 29 87
NOV 1 85
DEC 2 85
JAN 1 86
FEB 3 86
MAR 3 86
APR 1 86
MAY 1 86
JUN 2 86
JUL 1 86
AUG 1 86
SEP 1 86
OCT 1 86
NOV 3 86
DEC 1 86
JAN 1 87
FEB 2 87
MAR 2 87
APR 1 87
MAY 1 87
HP CONFIDENTIAL
FEB 24 86

REFERENCES

SECTION 8

[1] Military Standard, FTP, MIL-STD-1780, 10 May 1984

[2] Military Standard, SMTP, MIL-STD-1781, 10 May 1984

[3] Military Standard, Telnet, MIL-STD-1782, 10 May 1984

[4] Military Standard, IP, MIL-STD-1777, 12 August 1983

[5] Military Standard, TCP, MIL-STD-1778, 12 August 1983

[6] DDN X.25 Host Interface Specification, DCA, December 1983

[7] DDN Host Interface Qualification Testing, Link and Network Layers, DCA

[8] DDN Host Interface Qualification Testing, Transmission Control Protocol, Internet Protocol, DCA

[9] File Transfer Protocol, RFC765

[10] DoD Internet Host Table Specification, RFC952

[11] Internet Mail Protocols, SRI-NIC, November 1982

[12] DDN Compatibility for LAN/3000: Investigation Report, Dave Kasberg, August 23, 1985

[13] Official Arpa-Internet Protocols, RFC944

[14] An Ethernet Address Resolution Protocol, RFC826

[15] DDN Host Interface Qualification Testing, Higher-Level Protocols (Draft), DCA

[16] AdvanceNet Naming, Bob Carlson, December 16, 1984.

[17] File Transfer Protocol: Remote Driver Specification, System Development Corporation, 22 March, 1985.

[18] Telnet Protocol: Remote Driver Specification, System Development Corporation, 22 March 1985.

[19] TAC Users' Guide, Bolt, Beranek, and Newman Inc., December 1984.

Section 1
PRODUCT IDENTIFICATION

Section 2
PROBLEM STATEMENT

Section 3
MARKETING ANALYSIS

Section 4
INVESTIGATION RESULTS

Section 5
TELNET INVESTIGATION RESULTS

** END OF FORMATTING **
TDP/3000 (A.03.11) HP36578 Formatter
WED, JUL 9, 1986, 8:07 PM
NO ERRORS
INPUT = IR.PUB.DDN
OUTPUT = *HP2680

#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM

#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM

#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM

#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM
#J81; #O660 * IRJOB, KATY.DDN; SLP * WED, JUL 9, 1986, 8:09 PM