GC28-0673-1
File No. S370-37

Systems

OS/VS System Modification Program (SMP)

IBM

Second Edition (September, 1974)

This is a major revision of, and obsoletes GC28-0673-0. See the Summary of Amendments following the Contents. Changes or additions to text are indicated by a vertical line to the left of the change.

This edition applies to the System Modification Program (SMP) for OS/VS1 Release 3 and to OS/VS2 Release 2. This edition applies to all subsequent releases of OS/VS until otherwise indicated in new editions or Technical Newsletters. Changes are continually made to the information herein; before using this publication, in connection with the operation of IBM systems, consult the latest *IBM System/360 and System/370 Bibliography,* GA22-6822, for the editions that are applicable and current.

Requests for copies of IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for readers' comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Publications Development, Department D58, Building 706-2, PO Box 390, Poughkeepsie, N.Y. 12602. Comments become the property of IBM.

Preface

This publication contains the information necessary to install and use the System Modification Program (SMP). This information is intended for IBM personnel and system programmers responsible for the changing and redesigning of the installation's operating system.

The System Modification Program (SMP) provides the user with the facility to apply program temporary fixes (PTFs) or user modifications either selectively or as a group to an OS, VS1, or VS2 system or the distribution libraries (DLIBs) associated with them. The reader is expected to be experienced in using and modifying an OS or VS operating system.

This publication contains two chapters and an appendix:

Chapter 1: Introduction – contains a definition of SMP, describes the different ways that SMP can be used, and lists the hardware and software requirements. System generation considerations and storage requirements are also included. This chapter also contains a 'Flowchart of SMP Functions' with reference to topics in the manual that describe SMP functions.

Chapter 2: How to Use SMP – contains the requirements for using SMP, the operating instructions and JCL, as well as the SMP control statements with their format and parameters. Some typical examples are also included.

Appendix – contains detailed reference material for use with SMP as well as the messages that can be put out by SMP – their cause and origin and the corrective procedures that can be followed.

Associated Publications

SMP Reference Summary, GX28-0684
OS/VS System Modification Program (SMP) Logic, SY28-0685

The following publications may be required for reference during your use of SMP.

OS/VS Linkage Editor and Loader, GC26-3813
OS/VS and DOS/VS Assembler Language, GC33-4010
OS/VS Utilities, GC35-0005
OS/VS JCL Reference, GC28-0618
OS/VS Service Aids, GC28-0633

During the installation of SMP the following publications may provide needed data.

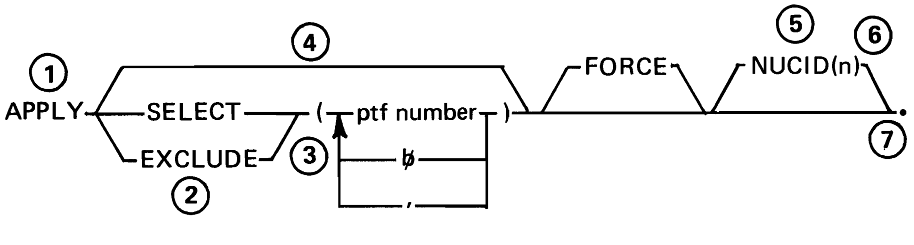
OS/VS1 System Generation, GC26-3791
OS/VS2 System Generation Reference, GC26-3792
OS/VS1 Storage Estimates, GC24-5094
OS/VS2 Storage Estimates, GC28-0604

Syntax Diagrams

A different notation for diagramming the syntax of the SMP control statements has been used. The basic rules that apply to these diagrams are:

- All upper case letters must be written as indicated.
- Words in lower case letters represent various statement parameters. An explanation of the parameters appears with each control statement.
- Any special character must be written as indicated.
- A split in the horizontal path indicates either an optional parameter or a choice of parameters at that point in the statement.
- Parameters that mav be reiterated are identified by a vertical line branching up and around them.
- Direction of flow is left to right unless a reiteration is indicated; a vertical line and an arrow indicates reiteration.
- Blanks can be used freely in an SMP statement. At least one blank must occur between each keyword (those in all capital letters) or each keyword and associated parameters.

The following example shows how the new syntax is used. All the rules described above are demonstrated here. To point out the relationships and rules, a numeric notation supplements the example.



1. The APPLY keyword is the only part of this statement other than the period that must appear.
2. Either the SELECT or the EXCLUDE keyword can be coded (but not both), and they must be accompanied by at least one PTF number.
3. The ptf number part of the statement must be replaced with a valid PTF number. More than one may be coded but they must be separated by a blank or a comma.
4. The SELECT and EXCLUDE keywords are optional and do not have to be coded.
5. The NUCID keyword is optional but if coded must have a number specified also.
6. The number may be coded with blanks between the keyword and the parenthesis, between the parenthesis and the number, between the number and the parenthesis.
7. The period is a special character and must be written as indicated.

Contents

Figures

Summary of Amendments for GC28-0673-1

This publication contains updated and clarified information for many areas of SMP. The major changes are:

SMP Data Set Names

Throughout the publication, the SMP data set names for the four main data sets are consistently shown as SMPCDS, SMPLOG, SMPPTS, and SMPACDS in all examples and figures. However, for ease of reading, the acronyms of CDS, LOG, PTS, and ACDS are used in the text material. (Note that the history log data set is shown as LOG or SMPLOG and not HLDS.)

SMP Examples

All examples have been updated with the correct SMP and system information.

System Requirements

The SMP storage and data set requirements have been updated. (See Chapter 1 "Storage Requirements".)

DATE=date Parameter

A new parameter is added to the EXEC statement for SMP. (See Chapter 2 "JCL for SMP".)

LIST Statement

Additional information is given for the LIST statement to describe (1) the contents of the listings for each type of CDS member, and (2) the messages produced by the LIST LOG statement. (See Chapter 2 "LIST".)

JCLIN Restrictions

The input restrictions for JCLIN processing are added. (See Chapter 2 "JCLIN".)

Update Control Language (UCL)

AC=1 and ALIGN2 are added under the LMOD linkage editor options. REGEN and XZAP are added under the PTF option. (See Chapter 2 "Update Control Language," figure 11.)

Data Set Descriptions

Additional information is provided for the SMP data sets, and 'lklib' and 'txlib' are described. (See Chapter 2 "Data Sets".)

PTF and ZAP Examples

Examples for the PTF and ZAP modification control statements are added. (See Chapter 2 "Handling User Modifications".)

PTF Tape Format and JCAMOD

Information about the PTF tape format and the JCAMOD program is removed. JCAMOD is no longer supported. (Refer to Chapter 2 "Program Temporary Fixes".)

Messages

Three new messages are added; HMA265I, HMA283I, and HMA399D. Also, other messages have added information. (See Appendix "Messages".)

Changes to the Appendix

The following changes have been made to the appendix:

- The 'Flowchart of SMP Functions' has been moved from the appendix to Chapter 1. It contains helpful references to the topics in the manual for the various SMP functions.
- The section 'Programming Considerations' has been removed from the appendix. Topics including this information have been added to the appropriate SMP functions in Chapter 2 under the heading 'Programming Considerations'.
- The section 'Cataloged Procedures' now has one suggested cataloged procedure for SMP.

Miscellaneous Changes

Throughout the publication, many clarifications and additions are made. The manual should be reviewed in its entirety.

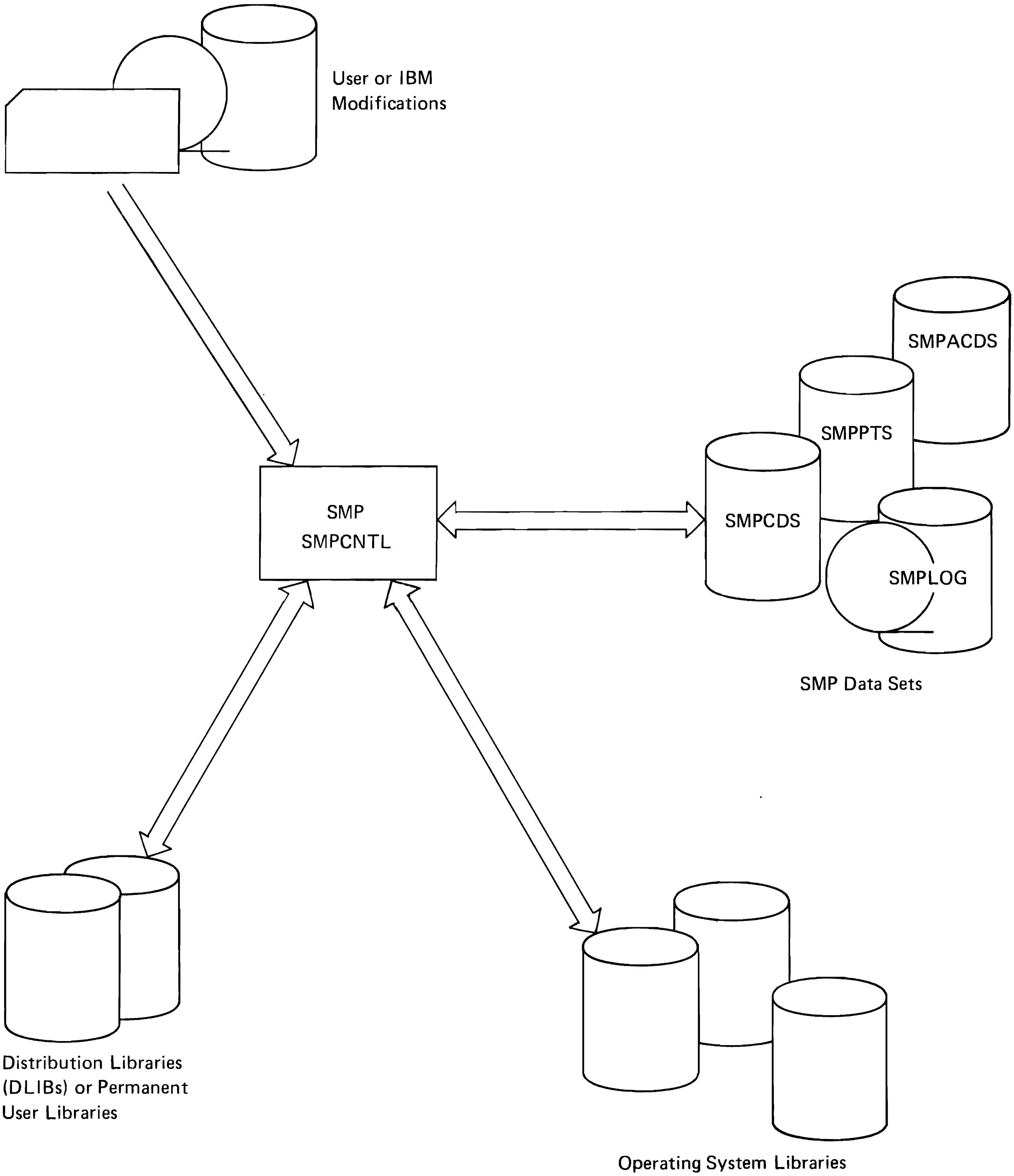

User or IBM Modifications
SMP
SMPCNTL
SMPACDS
SMPPTS
SMPCDS
SMPLOG
SMP Data Sets
Distribution Libraries (DLIBs) or Permanent User Libraries
Operating System Libraries

Figure 1. SMP Overview

Chapter 1: Introduction

It is often necessary to modify or change the software in a data processing system. These modifications or changes include patches for areas where there is a code problem and adding routines or programs to the system.

The System Modification Program (SMP) is a service aid that provides the facility to put IBM or user modifications into an OS, VS1, or VS2 system. The IBM modifications come on a tape that contains one or more modifications called program temporary fixes (PTFs). The user modifications can be module replacements, macro replacements, IMASPZAP control cards, or IEBUPDTE control cards and can be put in the system if they are prepared in the PTF format. An overview of SMP is provided in Figure 1 and shows the various data sets used during SMP execution.

Note that throughout the publication, references to IMASPZAP are for OS, HMASPZAP for VS1, and AMASPZAP for VS2.

The major functions SMP performs are:

- Checking input for accuracy and determining whether a particular modification applies to your system. This includes checking which system and release you are using and which options have been included and prerequisite and supercede verification.

- Putting IBM or user modifications in the operating system, distribution, or user libraries. The input can be handled all at once (as a group) or various modifications can be selected.

- Removing IBM or user modifications from the system. (This can be done as long as the distribution libraries – DLIBs – or permanent user libraries have not been updated by the modification being removed.)

- Maintaining records of the content and status of your operating system. A record of all the modules and macro instructions that make-up your system is created. (The macro instructions are initially the Stage II system generation macros.) Then, as modifications are made, records of these changes are added.

- Maintaining an audit trail of system modifications by means of time and date stamped records kept in a history log.

- Maintaining the status of modifications to the distribution library.

In addition, the contents of the input data (the modifications) or any of the SMP primary data sets can be printed. A user exit is provided so that user written routines can be executed during the checking process.

Purpose

The primary purpose of SMP is to put modifications, whether IBM or user, into an operating system. Or, conversely, SMP removes modifications that are not wanted or do not perform correctly. To accomplish these functions there are SMP control statements that you use to specify the action to be taken by SMP, some SMP data sets that must be defined, and certain system programs that must be in the system.

In preparing modifications to be put into a system, SMP first does extensive checking. This checking and verifying is done whenever the SMP control statement RECEIVE is included in an SMP job. Every modification must be checked and verified before anything else can be done by SMP for that modification. This ensures that only the modifications that affect your system configuration are used. The checking and verifying involves matching the system and release number of the modification with that of your system and prerequisite and supercede verification.

A record of the modules and macro instructions in your system is maintained by SMP on the control data set (CDS). If SMP determines that a modification can be put in a system, it makes an entry in the CDS with key control information and writes the input data to the PTF temporary store (PTS). During the execution of the major SMP functions the records on the CDS are used as a checklist of what is in the system and what modifications have been included. The entries in the CDS are used as a guide to the modifications that have been checked and are ready to be included in the operating system libraries, the DLIBs, or permanent user libraries.

The information contained in the CDS is essential to successful SMP operation – SMP cannot be run without the CDS. SMP is used to create the CDS from the Stage I output after each complete system generation. Then, after each partial system generation, SMP is used to update the CDS with the new Stage I output tape and two SMP procedures in SYS1.PROCLIB (SMPNUC and SMPIO).

There are four primary SMP data sets:

- CDS – (SMPCDS) Contains information about the present contents and status of your system.
- PTS – (SMPPTS) Is used to temporarily store modifications until they are placed on permanent storage. It contains the input data needed to make the modifications.
- LOG – (SMPLOG) Contains a history of each modification that has been made to your system. Each record, as it is written to SMPLOG, receives a time and date stamp.
- ACDS – (SMPACDS) Contains information about the present status of the distribution libraries (DLIBs).

SMP Functions

There are five major functions performed by SMP. These are directly concerned with the modification of a system. There are four support functions that are also part of the SMP service aid. These are used to create or update the CDS, process the Stage I output from system generation, and obtain listings of the CDS and the history log data set (LOG).

The control statements used for the five major SMP functions are:

RECEIVE – Checking and Verifying: All modifications must be checked or verified before any other SMP function can be executed. To do this the SMP control statement RECEIVE is used.

The data set, PTF Temporary Store (PTS), is updated during the checking and verifying operation. The update to PTS uses data contained in the Control Data Set (CDS) and the alternate control data set (ACDS) as controls. PTS is used to hold the modifications that are candidates for inclusion in the system. The CDS contains information about the modules and macro instructions that make-up your current operating system and is used to determine whether a particular modification should be included in your system. The information in PTS and CDS will be used during succeeding SMP executions.

A listing is also produced during the checking process. This listing contains all the modification control statements plus any messages generated during the processing.

REJECT – Deleting Modification Entries: If you decide not to use one or more of the modifications that you have processed with the RECEIVE statement, use the SMP control statement REJECT. This will delete the entries from PTS and CDS where they were written during the RECEIVE processing.

APPLY – Putting Modifications Into the System: The next procedure usually executed after the RECEIVE statement has processed the modifications, is putting the appropriate modifications into the operating system libraries of the most recent level of the system. This function is done using the SMP control statement APPLY. By changing the JCL DD cards and reissuing the APPLY, you may also put the modifications on a backup system.

RESTORE – Removing Modifications: If you want to remove any of the modifications put in the system with the APPLY control statement, you use the SMP control statement RESTORE. The modules and macro definitions that have been modified will be restored to status prior to this modification.

ACCEPT – Putting Modifications Into the DLIBs: When you are satisfied with the modified operating system, the modifications can be put into the distribution libraries (DLIBs) or permanent user libraries. The RESTORE function can no longer be used for these modifications. Use the SMP control statement ACCEPT for this function.

The statements used for the four SMP support functions are:

JCLIN – Creating the CDS: The JCLIN statement causes the Stage I output from system generation to be processed and a CDS to be created or updated.

UCLIN with ENDUCL – Updating the CDS: The UCLIN statement causes the statements that follow to be processed as UCL statements to update the CDS. The ENDUCL statement defines the end of the UCL statements.

LOG – Putting Messages on LOG: The LOG statement causes messages to be put on the history log data set (LOG).

LIST – Obtaining Listings: The LIST statement causes a listing to be produced from the control data set (CDS) or the history log data set (LOG).

System Requirements

SMP can be used on an OS, VS1, or VS2 system. SMP can be used on a minimum system, i.e., there are no special software or hardware requirements to use SMP.

Storage Requirements

There must be a minimum partition or region size of 96K to run SMP. (This assumes a minimum size is used for the linkage editor, assembler, etc.)

SMP resides in SYS1.LINKLIB along with the other service aids. The storage requirements for the program and the four SMP primary data sets are:

Device			**Tracks**		
	LINKLIB	**SMPCDS**	**SMPLOG**	**SMPPTS**	**SMPACDS**
2305	12	60	10	50	5
2311	48	240	40	200	20
2314/2319	24	120	20	100	10
3330/3333	13	66	11	55	6
3340	23	108	18	90	9

Data Set Requirements

The following data sets are used by SMP during the execution of SMP processing.

SMPCDS (CDS) This data set contains the records identifying the modules and macro instructions in your operating system as well as the status of PTFs and user modifications in your system.

SMPLOG (LOG) This data set contains time-stamped records of each modification that has been made to your system. Allocation for a 3300 can be SPACE=(CYL,(2,1)).

SMPPTS (PTS) This data set is used as temporary storage for the modifications until they are placed in the DLIBs or the permanent user libraries. Allocation for a 3300 can be SPACE=(CYL,(5,1,50)). A blocksize of 3200 is most efficient but can require more main storage for SMP processing. Note that directory entry information may be lost if TSO EDIT is used to modify PTF entries on the PTS.

SMPACDS (ACDS) This data set contains records that describe the status of PTFs and user modifications in your DLIBs. Allocate one track for every ten PTFs anticipated to be put into the DLIBs for the duration of the release. Allocate one directory block for each PTF anticipated that will be processed by the ACCEPT statement.

SYSPRINT This device is defined as SYSOUT; the data set definitions for SYSOUT are defined by the installation. The information printed contains the output for the system functions used by SMP (such as the linkage editor, assembler, and utilities).

SMPOUT This device is defined as SYSOUT; the data set definitions for SYSOUT are defined by the installation. The output from the processing of the SMP functions is printed here.

SYSUT1
SYSUT2
SYSUT3 Definition of these three data sets is based on assembler and utility requirements. These data sets are used for working data sets during SMP processing.

SYSLIB This data set is the macro library for the assembler – input only. These libraries are the same macro libraries, concatenated in the same sequence, as those that are used for the Stage II of system generation.

SMPMACn These data sets are also required during assemblies. There must be one of these for each of the assembler macro libraries. These are the libraries that will be updated by the modifications. They should be at least the Stage I and II system generation libraries as well as SYS1.MACLIB. There should be a one to one correspondence between the SMPMACn DD statements and the concatenated DD cards in the SYSLIB DD statement.

SMPPTFIN This data set contains the modification statements and the modifications; the IBM modifications are called program temporary fixes (PTFs).

SMPCNTL This data set contains the SMP control statement input.

SMPJCLIN This data set contains the Stage I output from system generation or other input in the same format.

tgtlib These data sets are the operating system libraries that will be modified.

distlib These data sets are the distribution libraries or permanent user libraries that will be modified.

lklib These data sets contain replacement modules that are in link edited format.

txlib These data sets contain replacement modules that are in object format and have not been link edited.

"Chapter 2: How To Use SMP" gives a detailed description of the purpose and use of each data set. The reference card contains a summary chart of when each data set is required. The summary chart can also be found in the appendix.

CDS Storage Estimates

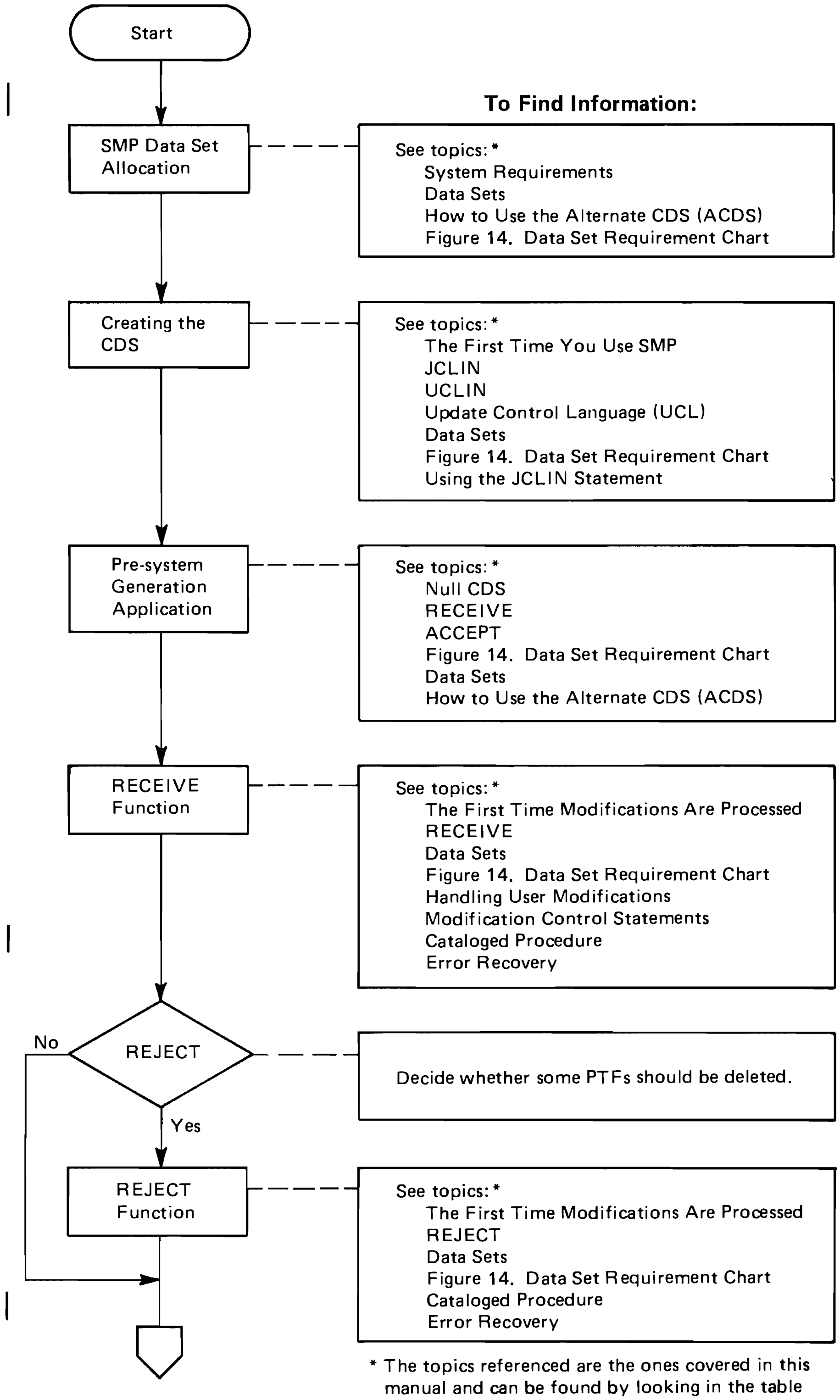
The amount of external storage required for the control data set (CDS) varies greatly with the system and options selected during system generation. In the same manner, the main storage required to create the CDS also varies. The minimum amount of main storage required is 96K, but the more main storage that is made available the faster the creation of CDS will run.

Since the number of modules in these systems is fairly constant, regardless of options selected during system generation, it is possible to provide a guideline for CDS allocation in relationship to storage size required for the VS systems. The following estimates are based on unblocked Stage I system generation output.

System	CDS Size	Number of Directory
	(2314)	Blocks
OS/VS1 (all releases)	4 cyls.	800
OS/VS2 (1.0.1.6)	5 cyls.	1000
OS/VS2 (2.0)	6 cyls.	1000

Also note that additional directory space must be allocated for PTF entries (at least 20 directory blocks). These additional blocks will not change CDS creation storage estimates.

Flowchart of SMP Functions (Part 1 of 2)



To Find Information:

See topics:*
System Requirements
Data Sets
How to Use the Alternate CDS (ACDS)
Figure 14. Data Set Requirement Chart

See topics:*
The First Time You Use SMP
JCLIN
UCLIN
Update Control Language (UCL)
Data Sets
Figure 14. Data Set Requirement Chart
Using the JCLIN Statement

See topics:*
Null CDS
RECEIVE
ACCEPT
Figure 14. Data Set Requirement Chart
Data Sets
How to Use the Alternate CDS (ACDS)

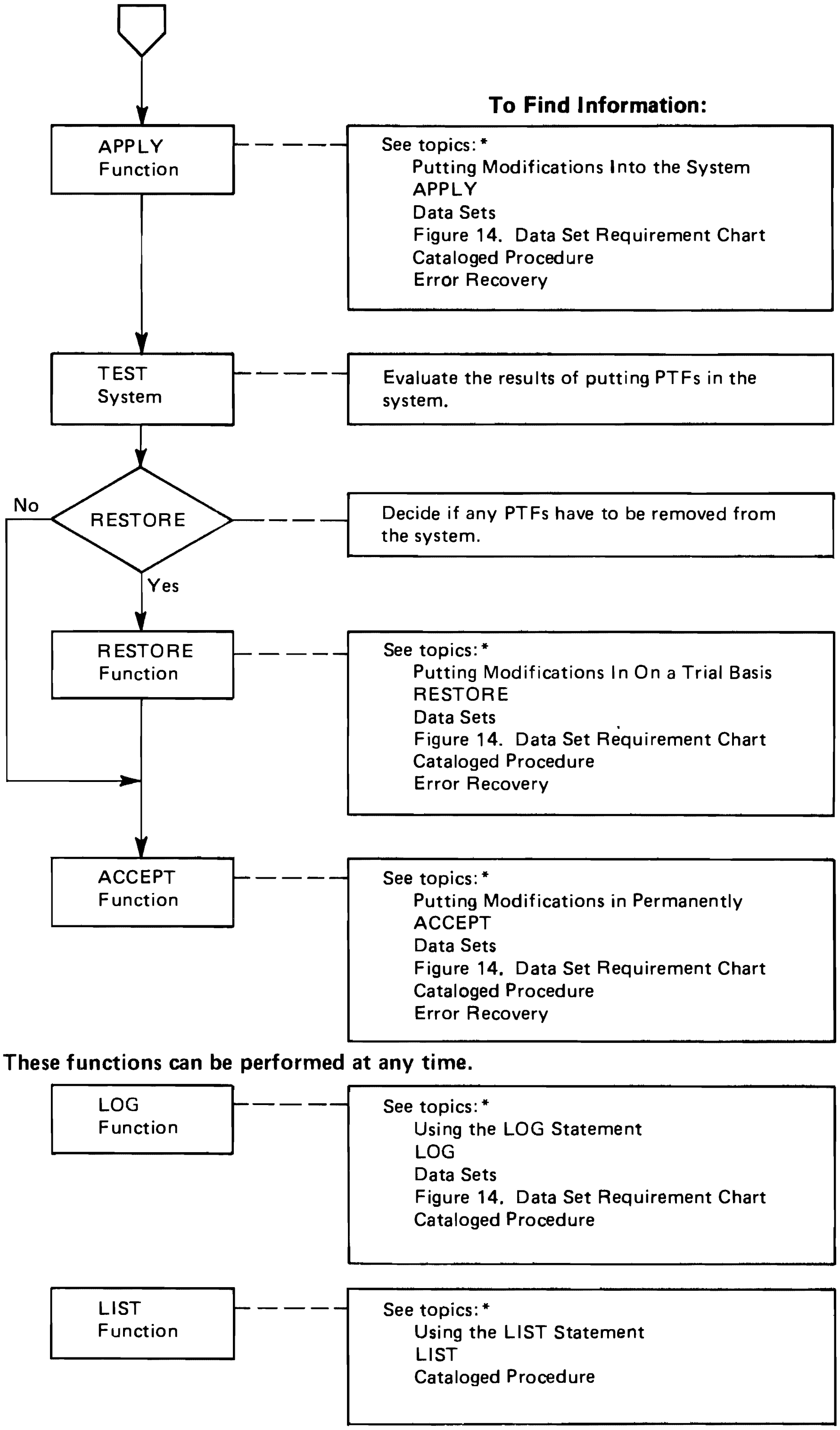
See topics:*
The First Time Modifications Are Processed
RECEIVE
Data Sets
Figure 14. Data Set Requirement Chart
Handling User Modifications
Modification Control Statements
Cataloged Procedure
Error Recovery

Decide whether some PTFs should be deleted.

See topics:*
The First Time Modifications Are Processed
REJECT
Data Sets
Figure 14. Data Set Requirement Chart
Cataloged Procedure
Error Recovery

* The topics referenced are the ones covered in this manual and can be found by looking in the table of contents.

Flowchart of SMP Functions (Part 2 of 2)



To Find Information:

APPLY Function: See topics:*
- Putting Modifications Into the System
- APPLY
- Data Sets
- Figure 14. Data Set Requirement Chart
- Cataloged Procedure
- Error Recovery

TEST System: Evaluate the results of putting PTFs in the system.

RESTORE: Decide if any PTFs have to be removed from the system.

RESTORE Function: See topics:*
- Putting Modifications In On a Trial Basis
- RESTORE
- Data Sets
- Figure 14. Data Set Requirement Chart
- Cataloged Procedure
- Error Recovery

ACCEPT Function: See topics:*
- Putting Modifications in Permanently
- ACCEPT
- Data Sets
- Figure 14. Data Set Requirement Chart
- Cataloged Procedure
- Error Recovery

These functions can be performed at any time.

LOG Function: See topics:*
- Using the LOG Statement
- LOG
- Data Sets
- Figure 14. Data Set Requirement Chart
- Cataloged Procedure

LIST Function: See topics:*
- Using the LIST Statement
- LIST
- Cataloged Procedure

* The topics referenced are the ones covered in this manual and can be found by looking in the table of contents.

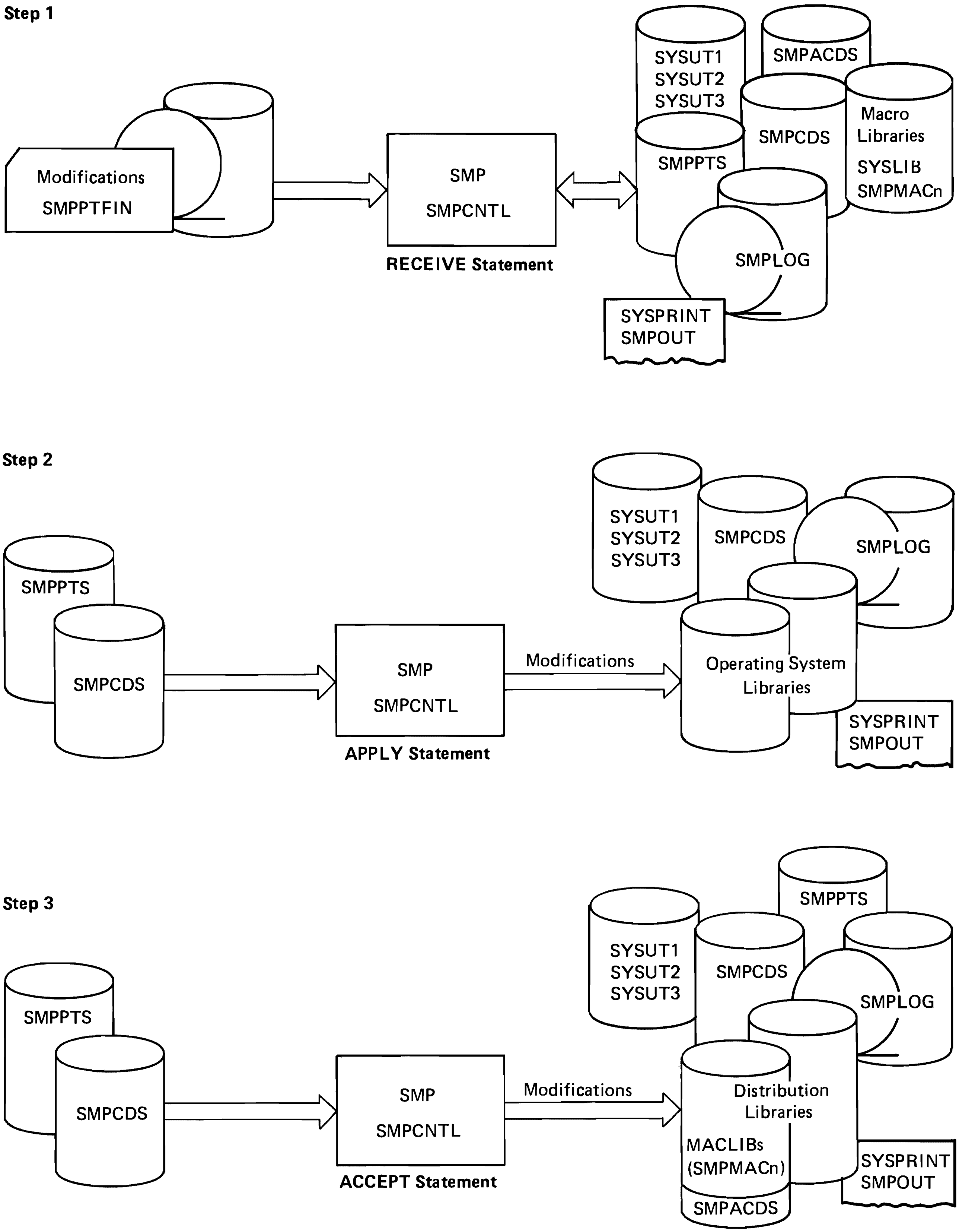


Figure 2. The Three Basic Steps for SMP

Chapter 2: How to Use SMP

The System Modification Program (SMP) puts IBM modifications, in the form of program temporary fixes (PTFs), and user modifications into the operating system, the distribution libraries (DLIBs), or the permanent user libraries. These modifications can affect any part of the system.

In order for SMP to determine which modifications should be put into a particular system, a data base containing information about this operating system must be created. The input for the creation of the data base is the Stage I system generation output. The SMP control statement used is JCLIN. The result is put into a data set called the control data set (CDS).

Changes to the CDS can be made using the SMP control statement UCLIN followed by control statements that provide the information about the change. These control statements are called the update control language (UCL) statements and are the method used to create records, change information, or delete data from the CDS. The end of the UCL statements is designated by the SMP control statement ENDUCL.

When modifications are processed by SMP, whether IBM or user, there are three basic steps or functions that are performed. Figure 2 shows these three basic steps.

The First Step – is the checking and verifying of the modifications for syntax errors, that the modification is applicable to this operating system, and that user modifications will not be overlaid. This is done with the SMP control statement RECEIVE. Each modification control statement, as it is encountered, is printed on the SMP data set SMPOUT. This listing provides a list of information about the modifications that have been processed and you can use it to determine the impact of applying the modification.

The RECEIVE function creates control records on the CDS and puts a record of all activity, time and date stamped, on a history log data set (LOG). In addition, the PTF temporary store (PTS) data set is used to hold modules and IMASPZAP (or HMASPZAP, AMASPZAP) control cards until this modification has been processed by the ACCEPT function. Any required assemblies are performed during RECEIVE processing and the resulting module is placed on the PTS.

During the checking done by the RECEIVE function certain conditions may not meet the criteria established in the RECEIVE function. When this happens a message is put out on SMPOUT and a flag called the force indicator is turned on. (See the detailed descriptions later in this chapter.) These modifications will not be put into the system unless they are specifically requested.

The listing produced during the RECEIVE processing can be used to select various modifications for inclusion in the system. The listing can also be used to determine which modifications are not wanted and the records representing these modifications can be removed. This deletion is done by using the SMP control statement REJECT.

The Second Step – is putting the modifications into the operating system libraries. This is done by using the SMP control statement APPLY. During this operation modifications that have the force indicator on can be put in by using the FORCE keyword. The system programs that may be required during this step are the linkage editor, IEBCOPY, IMASPZAP (or HMASPZAP, AMASPZAP), and IEHIOSUP. After the modifications have been put into the libraries the user should test the modified operating system.

If, during the testing, a modification does not work properly or you decide you don't want a particular modification in the system, these modifications can be removed. In order to remove the modifications, the module or macro instruction before the modification was put in, must be available to be put back. This back-off capability is provided by the SMP control statement RESTORE.

The Third Step – is putting the modifications into the DLIBs or the permanent user libraries. This is done by using the SMP control statement ACCEPT. This statement will also put the PTF information for the DLIBs into the alternate CDS (ACDS).

When this operation has been completed all the temporary records on the PTS have been deleted. There is no back-off capability for this modification after the ACCEPT processing is finished.

Other Functions – can also be performed by SMP. A listing can be obtained of the control data set (CDS), the alternate control data set (ACDS), or the history log data set (LOG) by using the SMP control statement LIST. You can also put data or messages into the LOG data set by using the SMP control statement LOG.

Using SMP

Using SMP involves specifying some JCL, some SMP control statements, and being sure you have all the required data sets set up and the system programs required available. There are examples throughout this chapter for each phase of using SMP.

Note: Be sure to substitute the correct DCB information, data set names, etc. that are appropriate for your installation.

The JCL required includes:

- EXEC statement with PGM=HMASMP plus any options that you decide to specify in the PARM keyword.
- DD statements for the various data sets required.

The SMP control statements used depend on the function or functions required. *Note:* Only one RECEIVE statement can be specified for each SMP job step. Don't forget RECEIVE must process each modification before any other SMP control statement can be used for that modification.

The data sets defined should include at least CDS, ACDS, PTS, and LOG plus any other input and output data sets needed. (See the data set requirements chart in the appendix.)

The system programs that should be in the system are the assembler, linkage editor, IEBCOPY, IEBUPDTE, IEHIOSUP, and IMASPZAP (or HMASPZAP, AMASPZAP).

SMP records modification activity in two forms – the control data set (CDS) maintains a record of the current status of the modifications in your operating system and the alternate control data set (ACDS) maintains the current status of your DLIBs. The history log data set (LOG) has a sequntial record of all modification activity that has taken place in the operating system libraries, the DLIBs, of the permanent user libraries.

JCL For SMP

The JCL statements required for SMP include the JOB and EXEC statements plus whatever DD statements are required for the functions you specify.

JOB Statement – Specify the installation dependent parameters.

EXEC Statement – You must specify PGM=HMASMP. You can specify in the PARM field:

ASM=name
DATE=date
LKED=name
ZAP=name
SIZE=(n1, n2)

ASM=name

specifies the name of the assembler. If none is specified, ASMBLR will be assumed.

DATE=date

specifies the date to be used:

U	Use IPL date of system
IPL	Use IPL date of system
REPLY	Request date from operator (message HMA399D)
yyddd	Use date supplied (yy = year, ddd = day)

If this parameter is not specified, then message HMA399D is used to request the date from the operator.

LKED=name

specifies the name of the linkage editor. If none is specified, IEWL will be assumed:

ZAP=name

specifies the name of the superzap program. If none is specified, IMASPZAP will be assumed.

SIZE=(n1,n2)

specifies the size of storage to be used by the linkage editor. If none is specified, default values for the linkage editor defined at system generation time will be used. (See the publication *Linkage Editor and Loader* for how to specify the size.)

The First Time You Use SMP

Before you can use SMP, the control data set (CDS) must be created. (An exception to this is described in the topic "Null CDS" later in this chapter.) This data set contains information about the modules, macro instructions, load modules, and assemblies that make-up your system. It is used as a guide and control during the application or removal of any modification. More detailed information about the CDS can be found under the topic "Data Sets" later in this chapter. Figure 3 shows the data sets used during the creation of the CDS and the direction of the data flow.

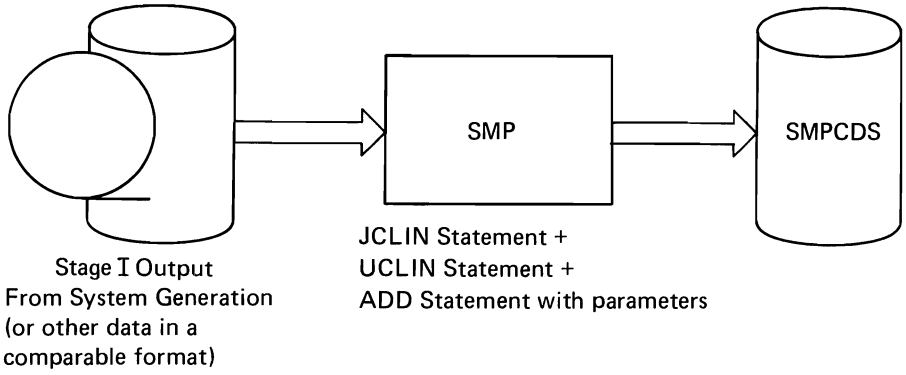


Figure 3. Creating the CDS

Creating the CDS – JCLIN and UCLIN Statements

To create the CDS you run SMP using two of the SMP control statements – JCLIN and UCLIN. The JCLIN statement causes SMP to read in the Stage I output from system generation and uses it to create or update the CDS. The UCLIN statement identifies the statements that immediately follow it as update control language (UCL) statements. The UCL statements are used to provide data to SMP that is used to make additions or changes to the CDS. The system type (OS, VS1, or VS2) and release level must be specified during the creation of the CDS. This is done with the UCL ADD statement. The system and release level information is used during the checking that is done during the RECEIVE processing.

The following example shows a typical control statement sequence to create the CDS and add the CDS system entry.

In the example, the input to JCLIN resides on a 2314 and SMP assumes preallocated CDS and LOG data sets.

```
//CDSCREAT JOB ACCT,NAME,MSGLEVEL=1
//S1  EXEC  PGM=HMASMP
//SMPOUT   DD SYSOUT=A
//SMPLOG   DD DSN=SMPLOG,DISP=MOD
//SMPCDS   DD DSN=SMPCDS,DISP=OLD
//SYSUT1   DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(10,10))
//SMPJCLIN DD DSN=STAGE2,UNIT=2314,VOL=SER=SMPVOL,DISP=OLD
//SMPCNTL DD *
    JCLIN .
    UCLIN .
    ADD SYS SREL (X020) NUCID (3) .
    ENDUCL .
/*
```

After this job has been run the CDS will contain an entry for each module, macro instruction, assembly, and load module in your system. There will also be a system entry, with information such as the type of operating system and release level, that will be used during the checking done by the RECEIVE routine (this entry contains information from the UCLIN data supplied). For each modification that is processed, a PTF entry is created and maintained with information about the status of the modification.

When another complete system generation is run from the same set of DLIBs or when multiple systems (when more than one system has been generated from the same set of DLIBs) are being used, refer to the topic "How to Use the Alternate CDS (ACDS)" in the appendix for a detailed description of the IEBCOPY step required.

Programming Considerations

- When executing SMP for the JCLIN function, if a small partition or region is used, creation time may become greater; therefore, the maximum space available should be allotted to SMP to perform this function.
- If insufficient directory blocks are allocated for the CDS, the need may arise to re-allocate, copy, and recreate the CDS.
- Partial system generations (such as IOGEN) must be input to SMP for the JCLIN function to insure that the CDS is kept current with module status. Note that the cataloged procedures for partial system generations must be executed prior to JCLIN.
- Whenever a system generation is performed and PTFs have been placed on the DLIBs, the ACDS must be copied *first* to the new CDS prior to JCLIN for the new system generation.
- Nucleus generation will always cause IEANUC01 CDS load module entry to be replaced. Therefore, if multiple nuclei are maintained simultaneously, more than one CDS will have to be used.

Null CDS

There may be an occasion when you want to run SMP before data has been put in the CDS from the Stage I output. For example, you may want to put PTFs into the DLIBs before the system is generated. The following example shows a typical control statement sequence to put modifications directly into the distribution libraries (AMODGEN, AMACLIB, AGENLIB, and AOSC5).

In the example, because this is a null CDS, no assemblies can be done. Therefore there is no need to specify any SYSLIB DD statements.

```
//NULLCDS    JOB  ACCT,NAME,MSGLEVEL=1
//STEP1      EXEC PGM=HMASMP
//SYSPRINT   DD   SYSOUT=A
//SMPOUT     DD   SYSOUT=A
//SMPLOG     DD   DSN=SMPLOG,DISP=MOD
//SMPACDS    DD   DSN=SMPACDS,DISP=OLD
//SMPCDS     DD   DSN=SMPCDS,DISP=OLD
//SMPPTS     DD   DSN=SMPPTS,DISP=OLD
//SMPMAC1    DD   DSN=SYS1.AMODGEN,DISP=OLD
//SMPMAC2    DD   DSN=SYS1.AMACLIB,DISP=OLD
//SMPMAC3    DD   DSN=SYS1.AGENLIB,DISP=OLD
//AOSC5      DD   DSN=SYS1.AOSC5,DISP=SHR
//SYSUT1     DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT2     DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSUT3     DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(1,1))
//SMPPTFIN   DD   UNIT=2400,VOL=SER=NLTAPE,DISP=OLD,                  *
//                LABEL=(2,NL),DCB=(RECFM=FB,LRECL=80,                *
//                BLKSIZE=3200)
//SMPCNTL    DD   *
   RECEIVE .
   ACCEPT FORCE .
/*
```

Note: The APPLY statement cannot be used until an entire CDS has been created. Therefore. the FORCE keyword is required on the ACCEPT statement to allow the modifications to be processed by ACCEPT without first being processed by APPLY. Also there will be no system and release checking unless you create the system entry using UCLIN followed with a UCL ADD statement with the appropriate parameters and ended with the ENDUCL statement.

Programming Considerations

- The CDS, PTS, LOG, and ACDS must be allocated.
- The system entry (SYS) in the CDS and ACDS should reflect the correct system and release.
- Space problems in the DLIBs can result and requires an occasional compress after ACCEPT processing.

The First Time Modifications Are Processed – RECEIVE Statement

When you are ready to put some modifications in your system, either some PTFs received from IBM or your own, set up an SMP job and use the RECEIVE control statement. The primary purpose of the RECEIVE routine is to check the modifications and prepare all the control information required by the other SMP functions. The RECEIVE control statement:

- Verifies the modifications.
- Identifies the modifications that apply to your system at that release level.
- Creates the CDS and PTS control entries required for future SMP processing.
- Prints a listing containing all the modification control records.
- Verifies prerequisite PTFs and user modifications and creates CDS and ACDS entries for superceded PTFs.

Figure 4 shows the data sets used as well as the data flow when the RECEIVE statement is being processed.

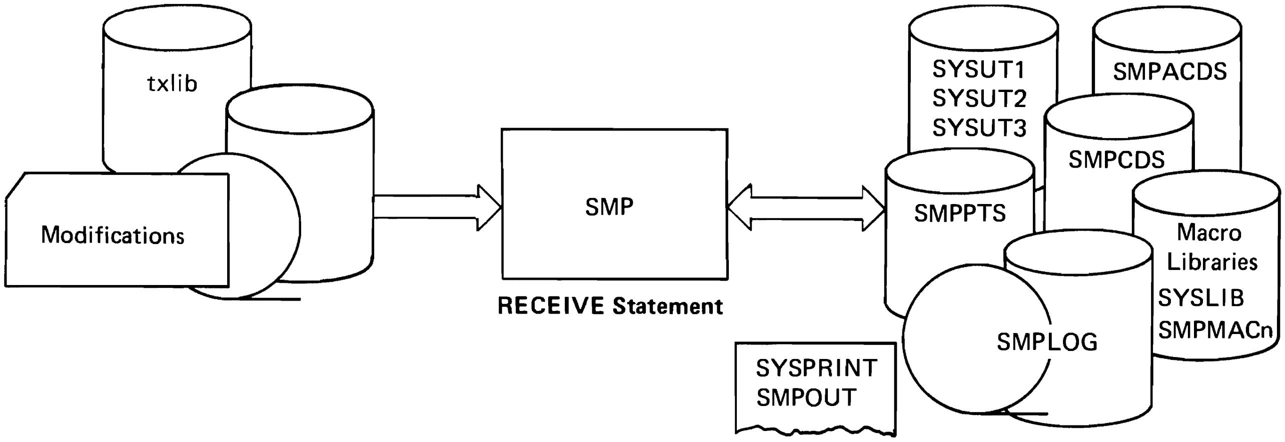


Figure 4. Processing Modifications

Processing

The RECEIVE processing must be the first SMP function executed for any modifications that are to be included in your operating system. For the modifications that apply to the type of system (OS, VS1, or VS2) but do not match in other respects such as release number, the modification is ignored. However, if you use the SELECT keyword, the RECEIVE routine will set a flag called the FORCE indicator. If prerequisites are not met, a warning message is issued and the FORCE indicator may be set.

An example of the control statements for a RECEIVE function follows.

```
//STEP1  EXEC PGM=HMASMP
//SMPOUT    DD   SYSOUT=A
//SMPLOG    DD   DSN=SMPLOG,DISP=MOD
//SMPCDS    DD   DSN=SMPCDS,DISP=OLD
//SMPPTS    DD   DSN=SMPPTS,DISP=OLD
//SMPACDS   DD   DSN=SMPACDS,DISP=OLD
//SMPPTFIN  DD   UNIT=2400,VOL=SER=PIDTAP,DISP=OLD,DSN=PIDTAP,       *
//               LABEL=(2,NL),DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200)
//SYSLIB    DD   DSN=SYS1.AMODGEN,DISP=SHR
//          DD   DSN=SYS1.AMACLIB,DISP=SHR
//SYSPRINT  DD   SYSOUT=A
//SYSUT1    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT2    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT3    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SMPMAC1   DD   DSN=SYS1.AMODGEN,DISP=SHR
//SMPMAC2   DD   DSN=SYS1.AMACLIB,DISP=SHR
//SMPMAC3   DD   DSN=SYS1.AGENLIB,DISP=SHR
//SMPCNTL   DD   *
   RECEIVE .
   LIST PTF /*list PTF status after RECEIVE */ .
/*
```

Examine the PTF documentation supplied with each shipment from IBM and the listing produced during the RECEIVE processing. Figure 5 shows a sample listing of the output produced during RECEIVE processing and a LIST PTF statement.

Programming Considerations

(1) Space problems:

- SYSUTn – the assembler requirements can be very large for modules assembled at SYSGEN. Check ASMS procedure for proper space values.
- PTS – mass RECEIVE can cause excessive space, therefore the PTS must have secondary allocation and should be blocked for maximum space efficiency (suggest BLKSIZE=3200).
- MACLIBS – these libraries may have to be enlarged if PTFs are to be maintained that have been processed by the RECEIVE and APPLY routines but not by ACCEPT, REJECT, or RESTORE.

(2) Pay close attention to messages put out from RECEIVE. Use the LIST PTF function following RECEIVE (and other SMP statements) to verify PTF status.

(3) The 'RECEIVE PROCESSING TERMINATED' message (rather than 'RECEIVE PROCESSING COMPLETED') indicates that more PTFs may exist on the PTF file at the point where the error occurred.

(4) PTFs may not be processed by RECEIVE simply because another PTF affecting the same modules or macros may have not been processed by ACCEPT. This causes modules to have copies on the PTS or macros to have back-up copies on the macro library.

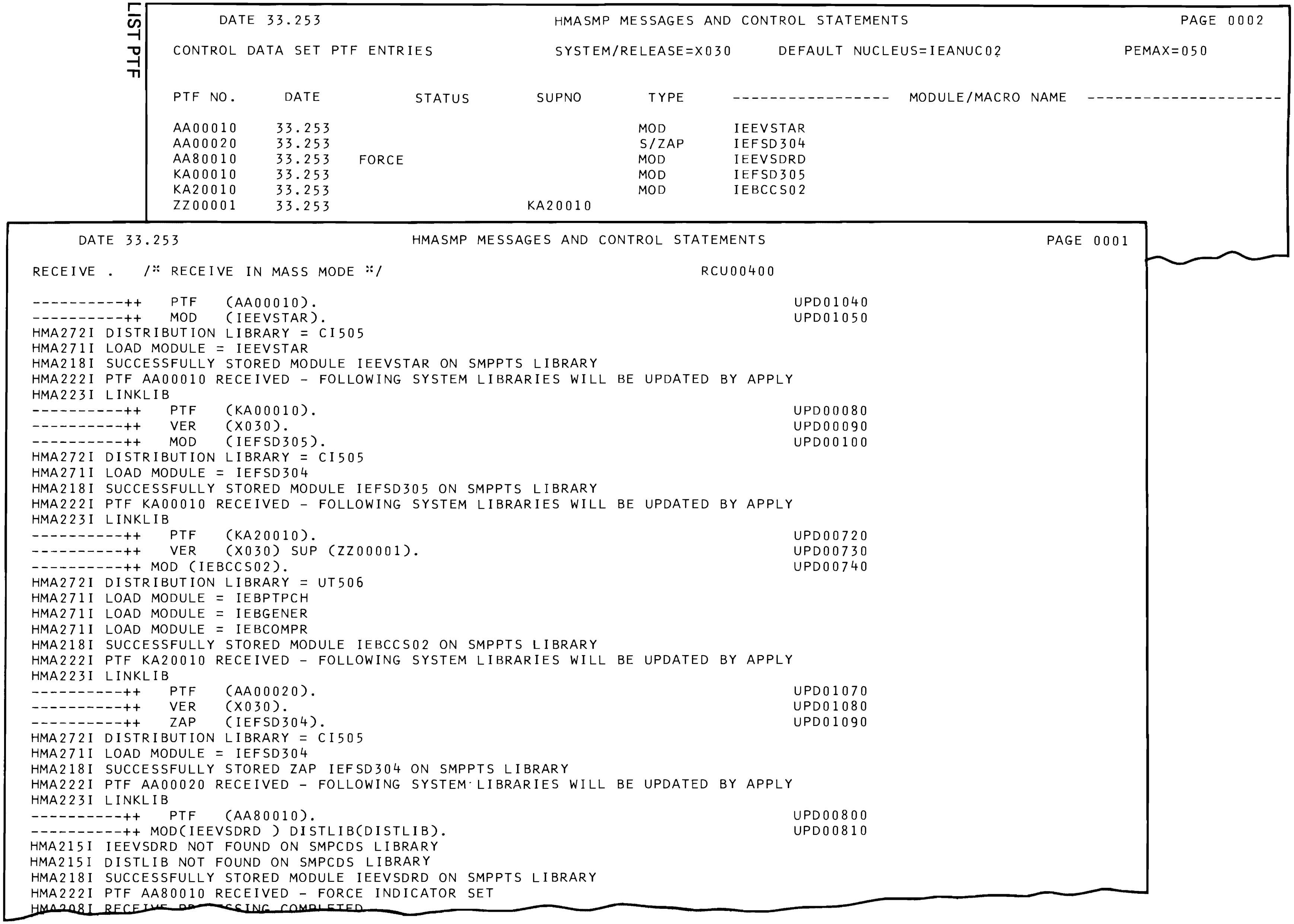
(5) The SMPMACn DD statements should allocate to all non-linked libraries which may be affected by PTFs (SAMPLIB, PROCLIB, RMTMAC, etc.).

(6) For macros which may cause reassemblies, there should be a DD card reflecting the macro's library in both SMPMACn and SYSLIB.

Detailed information about the RECEIVE statement can be found on the reference card as well as later in this chapter.

Removal of "Received Only" Modifications – REJECT Statement

You may want to delete entries from PTS and CDS for modifications before they are included in your system. This is done by using the REJECT control statement. Figure 6 shows the data sets used as well as the data flow during REJECT processing. You can remove all modifications that have been processed by a RECEIVE statement or you can select certain ones to be deleted. Note that the REJECT statement will delete entries that have been processed only by the RECEIVE statement.



LIST PTF

```
     DATE 33.253                             HMASMP MESSAGES AND CONTROL STATEMENTS                               PAGE 0002

  CONTROL DATA SET PTF ENTRIES               SYSTEM/RELEASE=X030     DEFAULT NUCLEUS=IEANUC02             PEMAX=050


  PTF NO.     DATE          STATUS       SUPNO       TYPE     -----------------  MODULE/MACRO NAME  ---------------------

  AA00010    33.253                                 MOD       IEEVSTAR
  AA00020    33.253                                 S/ZAP     IEFSD304
  AA80010    33.253   FORCE                         MOD       IEEVSDRD
  KA00010    33.253                                 MOD       IEFSD305
  KA20010    33.253                                 MOD       IEBCCS02
  ZZ00001    33.253                 KA20010
```

RECEIVE Processing

```
     DATE 33.253                             HMASMP MESSAGES AND CONTROL STATEMENTS                    PAGE 0001

  RECEIVE .   /* RECEIVE IN MASS MODE */                                     RCU00400

  ----------++   PTF   (AA00010).                                                      UPD01040
  ----------++   MOD   (IEEVSTAR).                                                     UPD01050
  HMA272I DISTRIBUTION LIBRARY = CI505
  HMA271I LOAD MODULE = IEEVSTAR
  HMA218I SUCCESSFULLY STORED MODULE IEEVSTAR ON SMPPTS LIBRARY
  HMA222I PTF AA00010 RECEIVED - FOLLOWING SYSTEM LIBRARIES WILL BE UPDATED BY APPLY
  HMA223I LINKLIB
  ----------++   PTF   (KA00010).                                                      UPD00080
  ----------++   VER   (X030).                                                         UPD00090
  ----------++   MOD   (IEFSD305).                                                     UPD00100
  HMA272I DISTRIBUTION LIBRARY = CI505
  HMA271I LOAD MODULE = IEFSD304
  HMA218I SUCCESSFULLY STORED MODULE IEFSD305 ON SMPPTS LIBRARY
  HMA222I PTF KA00010 RECEIVED - FOLLOWING SYSTEM LIBRARIES WILL BE UPDATED BY APPLY
  HMA223I LINKLIB
  ----------++   PTF   (KA20010).                                                      UPD00720
  ----------++   VER   (X030) SUP (ZZ00001).                                           UPD00730
  ----------++ MOD (IEBCCS02).                                                         UPD00740
  HMA272I DISTRIBUTION LIBRARY = UT506
  HMA271I LOAD MODULE = IEBPTPCH
  HMA271I LOAD MODULE = IEBGENER
  HMA271I LOAD MODULE = IEBCOMPR
  HMA218I SUCCESSFULLY STORED MODULE IEBCCS02 ON SMPPTS LIBRARY
  HMA222I PTF KA20010 RECEIVED - FOLLOWING SYSTEM LIBRARIES WILL BE UPDATED BY APPLY
  HMA223I LINKLIB
  ----------++   PTF   (AA00020).                                                      UPD01070
  ----------++   VER   (X030).                                                         UPD01080
  ----------++   ZAP   (IEFSD304).                                                     UPD01090
  HMA272I DISTRIBUTION LIBRARY = CI505
  HMA271I LOAD MODULE = IEFSD304
  HMA218I SUCCESSFULLY STORED ZAP IEFSD304 ON SMPPTS LIBRARY
  HMA222I PTF AA00020 RECEIVED - FOLLOWING SYSTEM LIBRARIES WILL BE UPDATED BY APPLY
  HMA223I LINKLIB
  ----------++   PTF   (AA80010).                                                      UPD00800
  ----------++ MOD(IEEVSDRD ) DISTLIB(DISTLIB).                                        UPD00810
  HMA215I IEEVSDRD NOT FOUND ON SMPCDS LIBRARY
  HMA215I DISTLIB NOT FOUND ON SMPCDS LIBRARY
  HMA218I SUCCESSFULLY STORED MODULE IEEVSDRD ON SMPPTS LIBRARY
  HMA222I PTF AA80010 RECEIVED - FORCE INDICATOR SET
  HMA208I RECEIVE PROCESSING COMPLETED
```

Figure 5. Sample Output From RECEIVE Processing and LIST PTF Statement

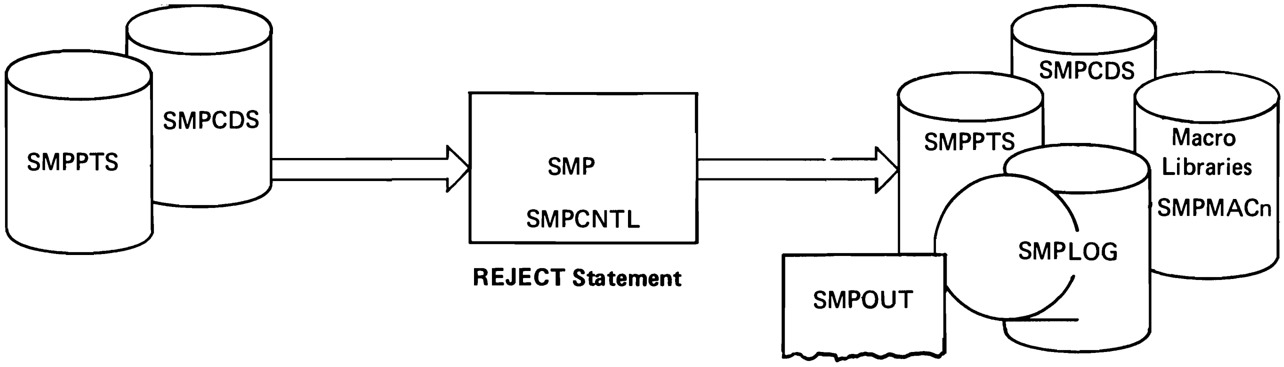


Figure 6. Removing Modifications Before Putting Them Into the System

```
//STEP1      EXEC PGM=HMASMP
//SMPOUT     DD   SYSOUT=A
//SMPLOG     DD   DSN=SMPLOG,DISP=MOD
//SMPCDS     DD   DSN=SMPCDS,DISP=OLD
//SMPPTS     DD   DSN=SMPPTS,DISP=OLD
//SMPMAC1 DD   DSN=SYS1.AMODGEN,DISP=OLD
//SMPMAC2 DD   DSN=SYS1.AMACLIB,DISP=OLD
//SMPMAC3 DD   DSN=SYS1.AGENLIB,DISP=OLD
//SMPCNTL DD   *
   REJECT .
/*
```

Programming Considerations

- It is suggested that the PTS and macro libraries be compressed after a mass REJECT.

Detailed information about the REJECT statement can be found on the reference card as well as later in this chapter.

Putting Modifications Into The Operating System

Modifications can be put in the operating system libraries on a trial basis or into the DLIBs (or permanent libraries) as a permanent change. The modifications can be handled in a group; for example, all the PTFs on the PTF tape you received from IBM, or, you can select various ones to be included. You should put any changes in on a trial basis first and test your operating system to be sure that the modifications are working correctly. After you have tested the modifications, then they can be put in the DLIBs or permanent libraries as a permanent change.

By the time the RECEIVE process finishes checking the modifications, entries will have been made in the PTS and the CDS for those modifications that can be included in your system. Those ready to be put into the system are the ones that were verified as accurate and that apply to your system, release level, and satisfied SMP prerequisites – everything else that applies to your system but does not completely check out has a FORCE indicator set. From the listing that is produced during the RECEIVE processing you can see what the impact of the modifications will be on your system. In addition, there are messages issued identifying missing prerequisite modifications and identifying modules that are contained in the incompatible modification.

Putting Modifications In On A Trial Basis – APPLY Statement

Modifications are placed in the operating system libraries by using the APPLY statement. You can put in everything, all modifications that have been processed by the RECEIVE function, or select one or more that you want. You can also, by specifying the FORCE keyword, put in modifications that have had the force indicator set during the RECEIVE processing. Figure 7 shows the data sets used as well as the data flow during APPLY processing.

Processing

To put all modifications into your system that have been processed by the RECEIVE function you only have to specify an APPLY statement without an operand. If you want to selectively put PTFs or modifications into your system, you use the APPLY statement specifying the SELECT keyword with the various PTF or modifications numbers coded. Either method can only be done after the modifications have been processed by the RECEIVE routine.

An example of the control statements for an APPLY function follows.

```
//STEP1 EXEC PGM=HMASMP
//SMPOUT   DD SYSOUT=A
//SMPLOG   DD DSN=SMPLOG,DISP=MOD
//SMPCDS   DD DSN=SMPCDS,DISP=OLD
//SMPPTS   DD DSN=SMPPTS,DISP=OLD
//LINKLIB   DD DISP=OLD,DSN=SYS1.LINKLIB
//SVCLIB   DD DISP=OLD,DSN=SYS1.SVCLIB
//NUCLEUS  DD DISP=OLD,DSN=SYS1.NUCLEUS
//SYSPRINT  DD SYSOUT=A
//SYSUT1   DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT2   DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT3   DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SMPCNTL  DD *
    APPLY .
/*
```

The listing produced by the RECEIVE routine listed the ddnames for the libraries LINKLIB, SVCLIB, and NUCLEUS as the libraries that are affected by the modification. Detailed information about the APPLY statement can be found on the reference card as well as later in this chapter.

Programming Considerations

(1) A space problem may result when considerable PTFs are put into the target system libraries with only the RECEIVE and APPLY (especially SYS1.NUCLEUS).

(2) PTFs are marked APPLY even if the return code from the linkage editor indicates a failure, therefore the linkage editor output must be scanned carefully on an error to determine whether or not the PTF actually was processed by the APPLY routine correctly.

(3) The ddnames required by SMP are the lowest level index name in the target library. (For example; use //SVCLIB for SYS1.SVCLIB.)

(4) If the TXLIB or LKLIB keywords are specified on the ++ MOD statement, the 'txlib' and 'lklib' DD cards must be specified for APPLY processing.

(5) The back-up capability of the nucleus may be lost if you use the same NUCID on two or more APPLY statements that affect the nucleus.

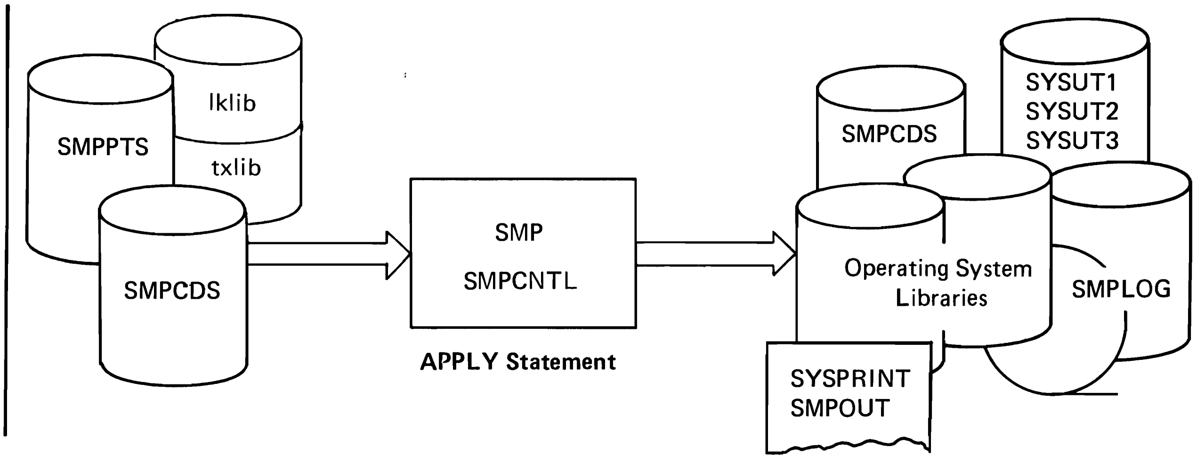


Figure 7. Putting Modifications Into the System on a Trial Basis

Removal of "Applied" Modifications – RESTORE Statement

You can remove any PTFs or modifications that you have put into the operating system libraries as long as they have not been processed by the ACCEPT function. This removal is done by using the RESTORE statement. Figure 8 shows the data sets used as well as the data flow during RESTORE processing. Saved or back-up copies of the modules and macro definitions are put back into the system libraries and then the PTFs and macro definitions are deleted from the PTS and CDS.

An example of the control statements for a RESTORE function follows.

```
//STEP1  EXEC  PGM=HMASMP
//SMPOUT   DD  SYSOUT=A
//SMPLOG   DD  DSN=SMPLOG,DISP=MOD
//SMPCDS   DD  DSN=SMPCDS,DISP=OLD
//SMPPTS   DD  DSN=SMPPTS,DISP=OLD
//LINKLIB  DD  DISP=OLD,DSN=SYS1.LINKLIB
//SVCLIB   DD  DISP=OLD,DSN=SYS1.SVCLIB
//NUCLEUS DD  DISP=OLD,DSN=SYS1.NUCLEUS
//SYSLIB   DD  DSN=SYS1.AMODGEN,DISP=OLD
//           DD  DSN=SYS1.AMACLIB,DISP=SHR
//SMPMAC1  DD  DSN=SYS1.AMODGEN,DISP=SHR
//SMPMAC2  DD  DSN=SYS1.AMACLIB,DISP=SHR
//SMPMAC3  DD  DSN=SYS1.AGENLIB,DISP=SHR
//SYSPRINT  DD  SYSOUT=A
//SYSUT1    DD  DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT2    DD  DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT3    DD  DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//AOSB3    DD  DSN=SYS1.AOSB3,DISP=SHR
//AOSC5    DD  DSN=SYS1.AOSC5,DISP=SHR
//SMPCNTL  DD  *
    RESTORE  S ( UX00203 ) .
/*
```

The ddnames for the libraries AOSB3 and AOSC5 were listed in the listing produced by the RECEIVE function as the distribution libraries affected by the modifications. Detailed information about the RESTORE statement can be found on the reference card as well as later in this chapter.

Programming Considerations

- Refer to the APPLY statement programming considerations numbers 1, 2, and 3.
- For macros which may cause reassemblies, there should be a DD card reflecting the macro's library in both SMPMACn and SYSLIB.

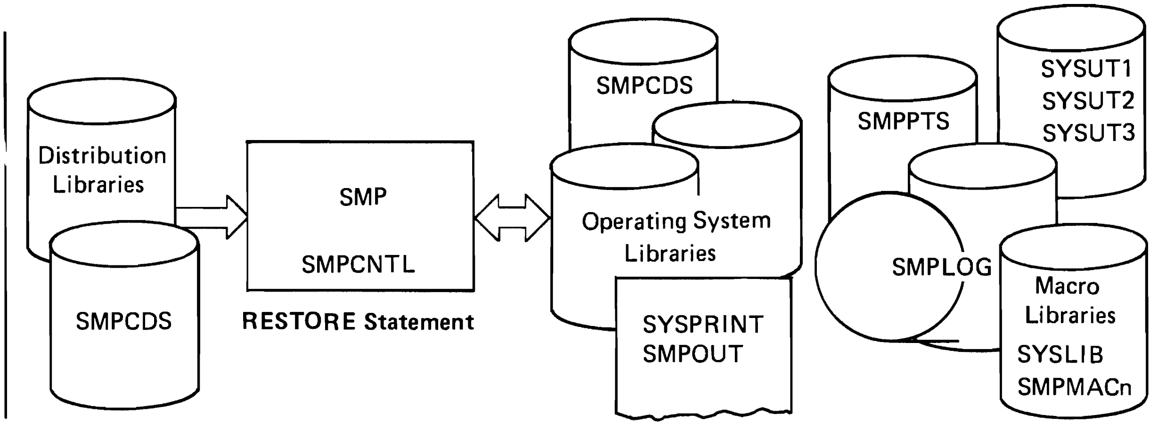


Figure 8. Removing Modifications From the System

Putting Modifications In Permanently – ACCEPT Statement

After you have tested out the PTFs and modifications, you will be ready to put them into the DLIBs or permanent libraries so that the next time you generate a system all these changes will be incorporated in the new operating system. This procedure is accomplished by using the ACCEPT statement. Figure 9 shows the data sets used as well as the data flow during ACCEPT processing. Once again, you can put the changes in as a group, select only certain ones, or force (with the FORCE keyword) any of the changes you want to be put in the libraries.

An example of the control statements for an ACCEPT function follows.

```
//STEP1     EXEC PGM=HMASMP
//SMPOUT    DD   SYSOUT=A
//SYSPRINT  DD   SYSOUT=A
//SMPLOG    DD   DSN=SMPLOG,DISP=MOD
//SMPCDS    DD   DSN=SMPCDS,DISP=OLD
//SMPACDS   DD   DSN=SMPACDS,DISP=OLD
//SMPPTS    DD   DSN=SMPPTS,DISP=OLD
//SMPMAC1   DD   DSN=SYS1.AMODGEN,DISP=OLD
//SMPMAC2   DD   DSN=SYS1.AMACLIB,DISP=OLD
//SMPMAC3   DD   DSN=SYS1.AGENLIB,DISP=OLD
//SYSUT1    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT2    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT3    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//AOSB3     DD   DSN=SYS1.AOSB3,DISP=SHR
//AOSC5     DD   DSN=SYS1.AOSC5,DISP=SHR
//SMPCNTL   DD *
   ACCEPT .
/*
```

When the ACCEPT processing has been completed the CDS entry or entries will be marked and a duplicate entry or entries will be created in the ACDS with the REGEN indicator set. The ddnames for the libraries AOSB3 and AOSC5 are found in the listing produced by the RECEIVE routine as the distribution libraries affected by the modification.

Be very careful because once these changes have been made to the DLIBs or the permanent user libraries the only way you can retrieve the former status of your operating system is to obtain a back-up copy of your DLIBs or permanent user libraries. Detailed information about the ACCEPT statement can be found on the reference card as well as later in this chapter.

Programming Considerations

CAUTION: This is a "point of no return" operation.

(1) Space problems can result on the distribution libraries when many PTFs have been processed by the ACCEPT routine. An occasional compress can regain space. ACCEPT may be re-issued if an ABEND occurred.

(2) PTFs are marked ACCEPT even if the linkage editor returns an error. Therefore, check the linkage editor output to verify that the PTF was processed properly by the ACCEPT routine. If not, ensure that the PTF entry is not on the ACDS and then; (1) correct the error and reprocess the PTF with ACCEPT, or (2) reprocess the PTF with RECEIVE and then ACCEPT with the FORCE keyword.

(3) When modules are link edited into the distribution libraries, external references are unresolved so ignore the message IEW046I.

(4) The ddnames required by SMP are the lowest level index name in the distribution library. (For example; use //SVCLIB for SYS1.SVCLIB.)

(5) If the TXLIB or LKLIB keywords are specified on the ++ MOD statement, the 'txlib' or 'lklib' DD cards must be specified for ACCEPT processing.

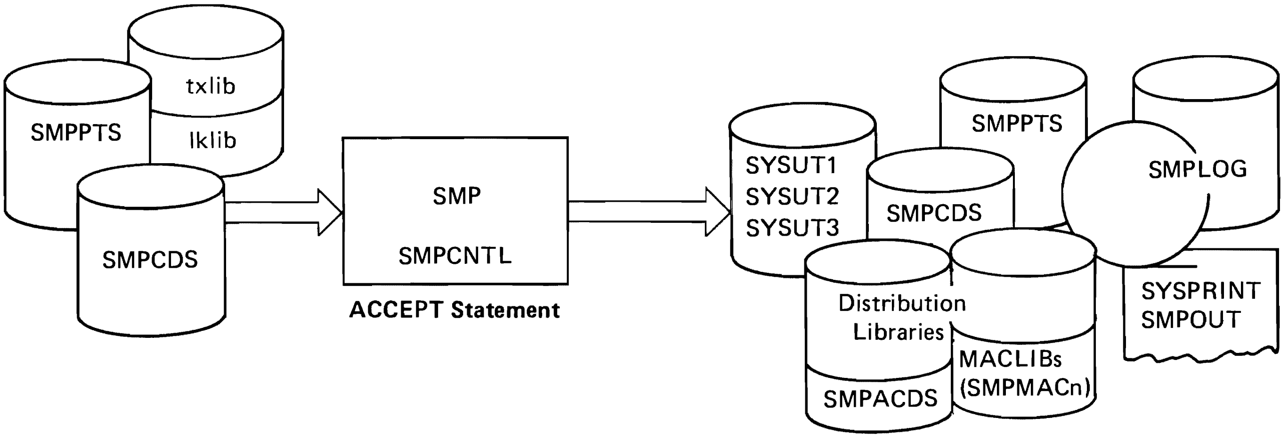


Figure 9. Putting Modifications Into the DLIBs for a Permanent Change

Other SMP Functions – Support Functions

There are four SMP support functions that can aid you during your use of SMP:

- JCLIN – This function causes the Stage I output from system generation (or other input in a similar format) to be read in and a CDS to be created or updated.
- UCLIN with ENDUCL – The UCLIN function causes the statements immediately following to be processed as UCL statements until an ENDUCL statement is encountered. UCL statements are used to update the CDS.
- LOG – This function causes a message to be written on LOG.
- LIST – This function causes a listing to be produced from CDS or LOG.

Using the JCLIN Statement

The JCLIN control statement causes SMPJCLIN to be read in as input. The CDS data set will be created or, if it exists, it will be updated.

The following statements can be used as an example of using the JCLIN statement. In the example, the input to JCLIN resides on a 2314 and SMP assumes preallocated CDS and LOG data sets.

```
//SGUPDAT JOB   ACCT,NAME,MSGLEVEL=1
//STEP1     EXEC PGM=HMASMP
//SMPOUT    DD   SYSOUT=A
//SMPLOG    DD   DSN=SMPLOG,UNIT=2314,VOL=SER=SMPVOL,DISP=MOD
//SMPCDS    DD   DSN=SMPCDS,UNIT=2314,VOL=SER=SMPVOL,DISP=OLD
//SMPJCLIN DD    DSN=STAGE2,UNIT=2314,VOL=SER=SMPVOL,DISP=OLD
//SYSUT1    DD   DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(10,10))
//SMPCNTL  DD    *
   JCLIN /* READ SYS GEN TAPE */ .
/*
```

Using the UCLIN Statement

The UCLIN control statement causes the statements that immediately follow it to be processed as update control language (UCL) statements. This continues until an ENDUCL statement is encountered.

The following statements can be used as an example of using the UCLIN statement.

```
//CDSUPDAT   JOB    ACCT,NAME,MSGLEVEL=1
//STEP1      EXEC   PGM=HMASMP
//SMPOUT     DD     SYSOUT=A
//SMPLOG     DD     DSN=SMPLOG,UNIT=2314,VOL=SER=SMPVOL,DISP=MOD
//SMPCDS     DD     DSN=SMPCDS,UNIT=2314,VOL=SER=SMPVOL,DISP=OLD
//SMPCNTL  DD       *
   UCLIN .
   ADD MAC (IEFCALL) ASSEM (IEFCALT) .
   ENDUCL .
/*
```

In the example, one of two actions are taken:

a. If the macro entry IEFCALL does not exist, the entry is created with the appropriate assembler entry (IEFCALT).

b. If the macro entry IEFCALL does exist, then IEFCALT is added to the assembler entries for IEFCALL.

Using the LOG Statement

The LOG control statement causes the operand to be treated as a message and put out on the history log data set (LOG).

The following statements can be used as an example of using the LOG statement.

```
//MSGOUT   JOB  ACCT,NAME,MSGLEVEL=1
//STEP1    EXEC PGM=HMASMP
//SMPOUT   DD   SYSOUT=A
//SMPLOG   DD   DSN=SMPLOG,UNIT=2314,VOL=SER=SMPVOL,DISP=(MOD,KEEP)
//SMPCDS   DD   DSN=SMPCDS,UNIT=2314,VOL=SER=SMPVOL,DISP=(OLD,KEEP)
//SMPCNTL  DD   *
   LOG (START OF MODIFICATIONS) .
/*
```

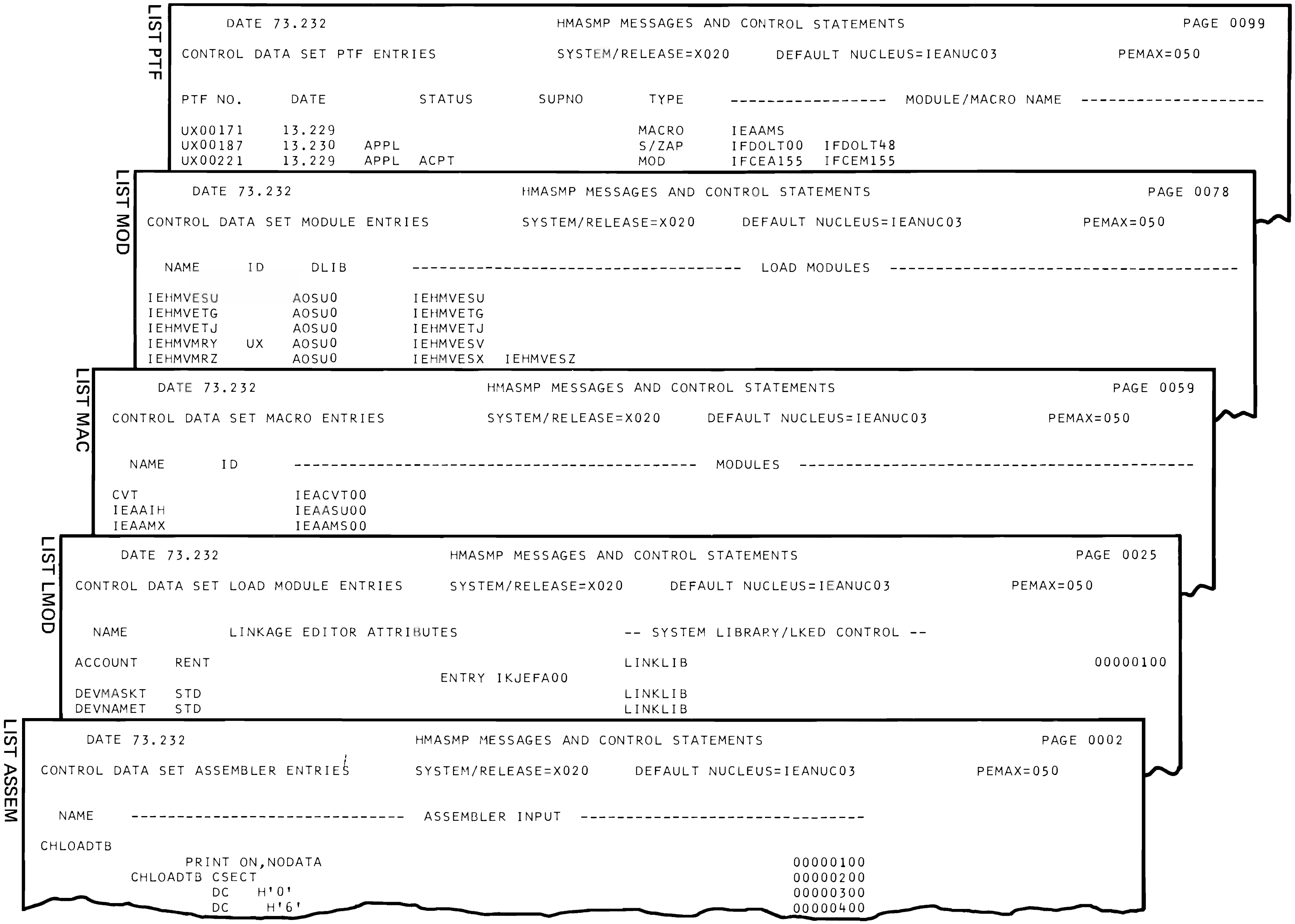
Using the LIST Statement

The LIST control statement causes a listing to be produced. The data can be selected types of entries from the CDS, the entire CDS, data within a date range from the LOG or the entire LOG data set. Figure 10 shows an example of a listing produced of CDS entries.

The following statements can be used as an example of using the LIST statement.

```
//LISTCDS  JOB  ACCT,NAME,MSGLEVEL=1
//STEP1    EXEC PGM=HMASMP
//SMPOUT   DD   SYSOUT=A
//SMPLOG   DD   DSN=SMPLOG,UNIT=2314,VOL=SER=SMPVOL,DISP=(MOD,KEEP)
//SMPCDS   DD   DSN=SMPCDS,UNIT=2314,VOL=SER=SMPVOL,DISP=(OLD,KEEP)
//SMPCNTL  DD   *
   LIST MOD MAC .
/*
```

In order to list the contents of the ACDS, specify the DD statement for SMPCDS to point to the alternate CDS (ACDS).



LIST PTF

```
     DATE 73.232                          HMASMP MESSAGES AND CONTROL STATEMENTS                                PAGE 0099

CONTROL DATA SET PTF ENTRIES              SYSTEM/RELEASE=X020      DEFAULT NUCLEUS=IEANUC03           PEMAX=050

PTF NO.      DATE          STATUS        SUPNO        TYPE       ----------------- MODULE/MACRO NAME -------------------

UX00171     13.229                                   MACRO      IEAAMS
UX00187     13.230    APPL                           S/ZAP      IFDOLT00   IFDOLT48
UX00221     13.229    APPL   ACPT                    MOD        IFCEA155   IFCEM155
```

LIST MOD

```
     DATE 73.232                          HMASMP MESSAGES AND CONTROL STATEMENTS                                PAGE 0078

CONTROL DATA SET MODULE ENTRIES           SYSTEM/RELEASE=X020      DEFAULT NUCLEUS=IEANUC03           PEMAX=050

  NAME      ID      DLIB        ---------------------------------- LOAD MODULES -------------------------------------

IEHMVESU          AOSU0        IEHMVESU
IEHMVETG          AOSU0        IEHMVETG
IEHMVETJ          AOSU0        IEHMVETJ
IEHMVMRY    UX    AOSU0        IEHMVESV
IEHMVMRZ          AOSU0        IEHMVESX   IEHMVESZ
```

LIST MAC

```
     DATE 73.232                          HMASMP MESSAGES AND CONTROL STATEMENTS                                PAGE 0059

CONTROL DATA SET MACRO ENTRIES            SYSTEM/RELEASE=X020      DEFAULT NUCLEUS=IEANUC03           PEMAX=050

  NAME       ID      ------------------------------------------ MODULES ------------------------------------------

CVT                  IEACVT00
IEAAIH               IEAASU00
IEAAMX               IEAAMS00
```

LIST LMOD

```
     DATE 73.232                          HMASMP MESSAGES AND CONTROL STATEMENTS                                PAGE 0025

CONTROL DATA SET LOAD MODULE ENTRIES     SYSTEM/RELEASE=X020      DEFAULT NUCLEUS=IEANUC03           PEMAX=050

  NAME           LINKAGE EDITOR ATTRIBUTES                 -- SYSTEM LIBRARY/LKED CONTROL --

ACCOUNT     RENT                                           LINKLIB                                        00000100
                                   ENTRY IKJEFA00
DEVMASKT    STD                                            LINKLIB
DEVNAMET    STD                                            LINKLIB
```

LIST ASSEM

```
     DATE 73.232                          HMASMP MESSAGES AND CONTROL STATEMENTS                                PAGE 0002

CONTROL DATA SET ASSEMBLER ENTRIES       SYSTEM/RELEASE=X020      DEFAULT NUCLEUS=IEANUC03           PEMAX=050

  NAME    ----------------------------- ASSEMBLER INPUT ------------------------------

CHLOADTB
                 PRINT ON,NODATA                                              00000100
          CHLOADTB CSECT                                                      00000200
                   DC    H'0'                                                 00000300
                   DC     H'6'                                                00000400
```

Figure 10. Sample Output From LIST Statements

SMP Control Statements

The SMP control statements are used to initiate an SMP function. The control statements are:

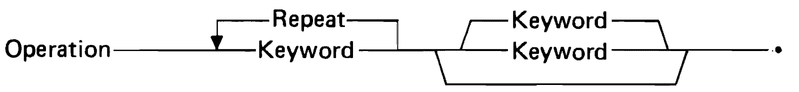
RECEIVE
REJECT
APPLY
RESTORE
ACCEPT
LOG
LIST
JCLIN
UCLIN
ENDUCL

Rules for Coding

1. SMP completes processing for one statement before it goes to the next. Therefore, the sequence of the statements determines the sequence of execution.
2. Periods indicate the end of each statement.
3. A statement continues until a period is encountered. This can be one or more records.
4. Each statement must begin on a new 80-column record.
5. Comments are delineated by /* at the beginning and */ at the end. A comment can appear anywhere within a statement before the ending period but cannot begin in column 1, if the data set is in the input stream.

Control Statement Format

The SMP control statements are written in a free-form style. They have the following format:



The statements can begin and end anywhere from column 1 through column 71. The operation must be specified first, followed by any keywords. The keywords can occur in any sequence. Each operation must begin a new record. Columns 73 to 80 are ignored by SMP; therefore, they can be used to sequentially number the records. A period (.) is used as an end of statement indicator.

Control Statements

A description of each SMP control statement is provided below. The information provided will enable you to:

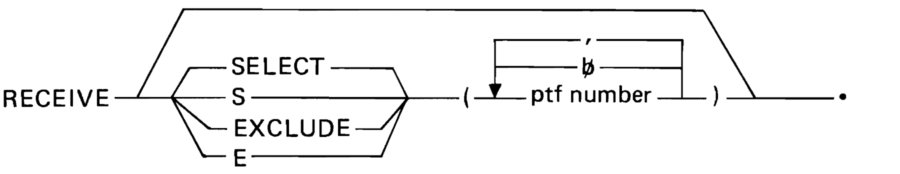
- Understand the purpose of each control statement.
- Know how to code each statement and the purpose of each keyword.
- Be aware of any special rules or restrictions.
- Know which data sets must be defined and which system programs must be in the system.

RECEIVE

The RECEIVE statement causes SMP to read and verify an input file of modifications. Only one RECEIVE statement can be included in an SMP job step.
Each PTF or user modification to be handled by SMP must be processed with a RECEIVE statement before any other SMP control statement can be used. It does preliminary processing such as:

- Checking for entries that apply to your system and release level.
- Checking the syntax of various statements.
- Checking for valid prerequisites and superceded PTFs.
- Modifying macros in the SMPMACn libraries.
- Performing assemblies as needed.
- Creating entries in CDS, LOG, and PTS.

All applicable PTFs or user modifications can be handled or certain modifications can be selected or excluded. An example of the statements you can use to specify the RECEIVE operation is provided earlier in this chapter under the topic "The First Time Modifications Are Processed."



SELECT (ptf number)

specifies the PTFs and user modifications that are to be processed from the input PTF or modification file. SELECT can be coded as S.

EXCLUDE (ptf number)

specifies the PTFs and user modifications that are not to be processed from the PTF input file. All others will be processed. EXCLUDE can be coded as E.

Note: All PTF header statements are passed to SMPOUT even though they are not selected for RECEIVE.

Rules

1. When all modifications on SMPPTFIN are to be processed by a mass RECEIVE, that is S or SELECT has not been specified, the PTF or user modification must:
   - Apply to this operating system at this release level. (The system and release information (SREL) field in the CDS system entry must match the system and release information in the modification control statement VER.)
   - Not already exist (have an entry) in the PTF section of the CDS.

   Note: If SELECT is specified, rule 1 does not apply.

2. For a mass RECEIVE and selective RECEIVE, the PTFs and user modifications are acceptable to the RECEIVE process routine only if:
   - The modules do not have existing entries in the PTS.
   - The macro definitions do not have saved copies in the macro library.
   - A macro instruction will not cause re-assembly of a module that has an existing entry in the PTS.

3. The FORCE indicator is set if:
   - SELECT is specified and the PTF or user modification does not apply to this operating system at this release level.
   - SELECT is specified and the PTF or user modification already exists in the PTF section of the CDS.
   - The prerequisite PTF has not been processed by the RECEIVE routine or has only been processed by RECEIVE.
   - The modification identifiers in the CDS for all the modules and macro definitions affected by this PTF or user modification do not match the identifier for this PTF or user modification. (See the description of the modifiers in the topic 'Modification Control Statements' later in this chapter.)

4. If the prerequisite has been processed by the APPLY routine only or the ACCEPT routine only, message HMA212I will indicate the status of the prerequisite PTF but the FORCE indicator will not be set. Note that if the prerequisite and the major hit the same module, the major PTF is flushed due to rule number 2.

Data Sets Required for RECEIVE

To use the RECEIVE statement, the following data sets are required:

SMPCDS (CDS)
SMPACDS (ACDS)
SMPPTS (PTS)
SMPPTFIN
SYSPRINT
SMPOUT
SMPLOG (LOG)
SYSUT1
SYSUT2
SYSUT3
SYSLIB
SMPMACn
SMPCNTL

Notes:
1. SMPACDS should not be on-line if you are receiving a modification to a non-sysgen macro and the modification has already been accepted.
2. SMPMACn and SYSLIB are only required when processing ++ MAC or ++ UPDTE modifications.

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

After RECEIVE statement processing:

- All modules and superzaps are stored on the PTS.
- Macros are updated on the macro libraries and a back-up copy is saved.

Printed Output

A listing (SMPOUT) is produced during the processing initiated by the RECEIVE statement. It contains all the modification control statements for those PTFs selected plus any messages the RECEIVE processing generates.

This gives you the chance to evaluate the impact of the modification before it is actually included in the system.

All load module names associated with a particular module are listed as are the distribution libraries where the modules are located. All unique system libraries that will be updated are listed after the RECEIVE function is finished processing a modification.

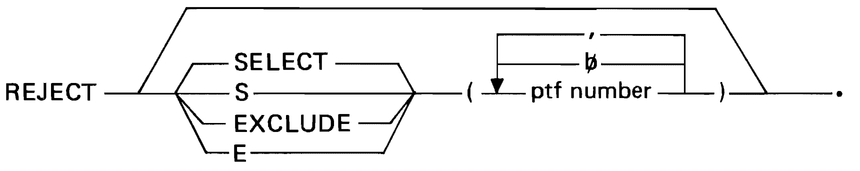
Other Programs Required for RECEIVE

The RECEIVE statement may cause one or more of the following programs to be executed:

IEBUPDTE
Assembler

REJECT

The REJECT statement causes any PTFs or modifications that have been processed only by the RECEIVE statement to be deleted. More than one REJECT statement can be included in an SMP job. An example of the statements you can use to specify a REJECT operation is provided earlier in this chapter under the topic "The First Time Modifications Are Processed."



SELECT (ptf number)

specifies the PTFs and user modifications that are to be deleted. SELECT can be coded as S.

Note: If SELECT is not specified, all PTFs and user modifications that have only been processed by the RECEIVE statement will be deleted.

EXCLUDE (ptf number)

specifies the PTFs and modifications that are not to be deleted. All others will be deleted. EXCLUDE can be coded as E.

After the REJECT statement processing:

- All modules and superzaps are deleted from PTS for those PTFs rejected.
- Updated macro definitions are deleted from the macro libraries and the old copy is restored (SMPMACn).
- PTF entries in the CDS are deleted.

Data Sets Required for REJECT

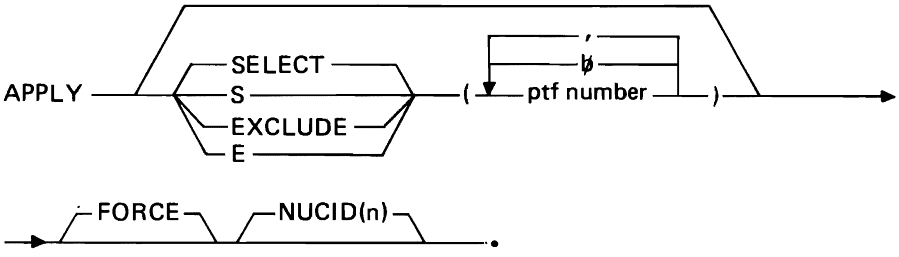
To use the REJECT statement the following data sets are required:

SMPOUT
SMPLOG (LOG)
SMPCDS (CDS)
SMPPTS (PTS)
SMPMACn
SMPCNTL

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

APPLY

The APPLY statement causes PTFs and user modifications processed by a RECEIVE statement to be put into the operating system libraries. (This statement does not cause any changes in the distribution – DLIB – or permanent user libraries.) There can be any number of APPLY statements in an SMP job. An example of the statements you can use to specify an APPLY operation is provided earlier in this chapter under the topic "Putting Modifications In On A Trial Basis."



If no operand is specified, all PTFs and user modifications that have not been processed by an APPLY or ACCEPT statement and have not had the force indicator turned on during the RECEIVE processing will be put into the system.

SELECT (ptf number)

specifies the PTFs and user modifications that are to be put into the operating system. SELECT can be coded as S.

EXCLUDE (ptf number)

specifies the PTFs and user modifications that are not to be put in the operating system. All others will be processed. EXCLUDE can be coded as E.

FORCE

must be specified if you want the PTF and user modifications that have the FORCE indicator set to be put in the operating system.

Note: When FORCE is specified, it applies to all the PTFs and user modifications processed by that APPLY statement.

NUCID(n)

n is a number that is put on the end of IEANUC0 to form a name under which the current nucleus is saved when the nucleus is to be updated by this APPLY statement. This parameter overrides the NUCID parameter that was specified when the system release ID was created. This overriding is done only for this APPLY statement.

Notes:

1. If the nucleus is updated and NUCID is not specified, the default nucleus ID in the CDS system entry is used. If neither is present, a copy of the nucleus before the change will not be made.
2. The current nucleus is not saved if the PTF is a superzap.
3. If a module in a PTF is not found on the CDS and it is not in a totally copied library, that module will not be processed by the APPLY statement.

Data Sets Required for APPLY

To use the APPLY statement the following data sets are required:

SYSPRINT
SMPOUT
SMPLOG (LOG)
SMPCDS (CDS)
SMPPTS (PTS)

SYSUT1
SYSUT2
SYSUT3
SMPCNTL
tgtlib

The data sets 'lklib' and 'txlib' are required when the TXLIB or LKLIB keywords are specified on the ++ MOD statement.

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

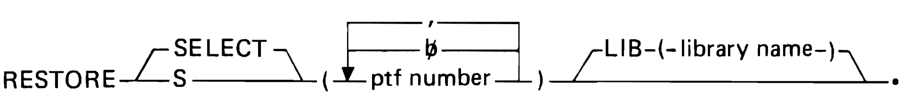
Other Programs Required By APPLY

The APPLY statement may require one or more of the following programs to be called:

IMASPZAP
Linkage Editor
IEBCOPY
IEHIOSUP (for OS and VS1 if SVCLIB was updated). See appendix for detailed description of executing IEHIOSUP.

RESTORE

The RESTORE statement causes the PTFs or user modifications processed by the APPLY statement to be removed from the operating system. The versions of the modules that were in use before the change are put into the operating system libraries. Saved copies of macro definitions on the macro library and any required modules are reassembled and are put into the operating system libraries. All modules that were superzapped are re-linkedited with copies from the DLIBs. An example of the statements you can use to specify a RESTORE operation is provided earlier in this chapter under the topic "Putting Modifications In On A Trial Basis."



SELECT (ptf number)

specifies the PTFs and user modifications that are to be removed from the system. SELECT can be coded as S.

LIB (library name)

specifies the ddname of the permanent user library to be used. This is specified if the DLIBs do not contain the data for the correct module level to be used in restoring the operating system libraries.

Each PTF and user modification to be removed must be specified by SELECT. Only those that have been processed by the APPLY function and not by the ACCEPT function will be processed.

After RESTORE statement processing:

- The module that was being used before the change is retrieved from the DLIBs or, if the LIB keyword was specified, from a permanent user library.
- The updated or changed macros are deleted from the proper macro library and the old copy is retrieved.
- After each PTF or user modification has been restored to its previous version, the CDS entry is deleted and all modules affected are deleted from the PTS.

Data Sets Required for RESTORE

To use the RESTORE statement the following data sets are required:

SYSPRINT
SMPOUT
SMPLOG (LOG)
SMPCDS (CDS)
SMPPTS (PTS)
SYSUT1
SYSUT2
SYSUT3
SYSLIB
SMPMACn
SMPCNTL
tgtlib
distlib

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

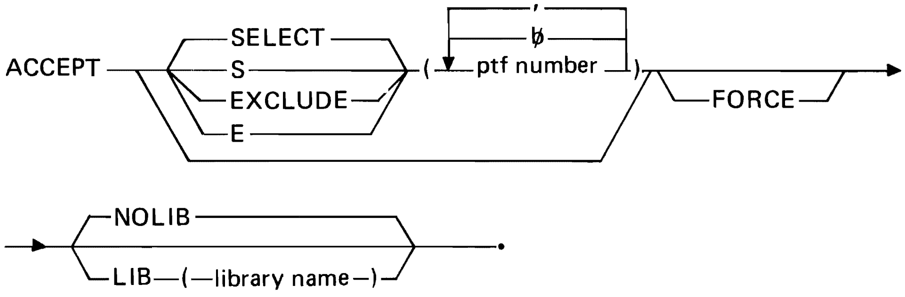
Other Programs Required for RESTORE

The RESTORE statement may require one or more of the following programs to be called:

IEBCOPY
Linkage Editor
Assembler
IEHIOSUP (See appendix for detailed description for executing IEHIOSUP.)

ACCEPT

The ACCEPT statement causes the modifications processed by an APPLY statement to be put into the DLIBs or a permanent user library. There can be any number of ACCEPT statements in an SMP job step. Once this statement has processed a modification, SMP cannot remove it from your system. An example of the statements you can use to specify an ACCEPT function is provided earlier in this chapter under the topic "Putting Modifications in Permanently."



If no operand is specified, all the PTFs or user modifications that have been processed by an APPLY and not an ACCEPT function will be put into the DLIBs or permanent user libraries.

SELECT (ptf number)

specifies the PTFs and user modifications that are to be put into the DLIBs or the permanent library specified in the LIB keyword. SELECT can be coded as S.

EXCLUDE (ptf number)

specifies the PTFs and user modifications that are not to be put in the DLIBs or the permanent libraries specified in the LIB keyword. EXCLUDE can be coded as E.

Note: SMP will ACCEPT any PTFs that have the FORCE indicator on (and not specified by EXCLUDE).

FORCE

specifies that all PTFs and user modifications that have been processed by the APPLY or RECEIVE statements are to be put into the DLIBs or permanent user libraries.

Notes:

1. The FORCE keyword used in the ACCEPT statement is not related to the force indicator in the CDS PTF entry. It merely allows the bypass of the APPLY function.
2. If SELECT and FORCE keywords are specified, all the selected modifications processed by RECEIVE, APPLY, or ACCEPT statements will be put into the DLIBs or the permanent libraries.

LIB(library name)

specifies the permanent user library to be used instead of the DLIBs for the modifications specified in the ACCEPT statement in which this keyword is coded. Use the ddname of the library specified in this job stream that you want to update.

Note: If IMASPZAP is required during the processing of a modification and the LIB keyword is specified, the user is responsible for providing a copy of the module to be updated in the named library. If you do not provide the copy of the module, this permanent update in the specified library will be lost.

NOLIB

specifies that regular ACCEPT processing will occur but that no update will be made to the permanent libraries. To zap assembled modules, the NOLIB keyword should be used because the module does not exist on a DLIB data set.

After a PTF or modification has been processed by the ACCEPT statement, the following conditions will exist:

1. The saved copies of macro definitions are deleted from the macro library (SMPMACn); only the updated version remains. The PTS entries are deleted for modules, superzaps, and assembled modules for the updated macro instruction.
2. The modification identifier in the CDS module and macro entries has been changed to that of the PTF or user modification just processed.
3. The PTF information for the DLIBs has been recorded in the ACDS.
4. The REGEN indicator is set in the PTF entry in the ACDS. After the PTF is placed into the DLIBs, if a system generation is performed and the ACDS is copied to the CDS, the REGEN indicator in the CDS indicates to the re-generated system that this PTF may now be in the system. (This assumes that the system has been generated with the options that link the modules affected by this PTF.)

Data Sets Required for ACCEPT

To use the ACCEPT statement the following data sets are required:

SYSPRINT
SMPOUT
SMPLOG (LOG)
SMPCDS (CDS)
SMPPTS (PTS)
SMPACDS (ACDS)
SYSUT1
SYSUT2
SYSUT3
SMPMACn
SMPCNTL
distlib

The data sets 'txlib' and 'lklib' are required when the TXLIB or LKLIB keywords are specified on a ++ MOD statement.

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

Other Programs Required by ACCEPT

The ACCEPT statement may require one or more of the following programs to be called:

Linkage Editor
IEBCOPY
IMASPZAP

LOG

The LOG statement causes a message to be placed in the LOG data set. An example is provided earlier in this chapter under the topic "Other SMP Functions."

```
LOG ———— ( ———— message ———— ) ————•
```

message

is written in the LOG data set along with the time and date that the message was generated. Any character can be used between the parentheses. If parentheses are used, they must be balanced.

Messages generated by the various SMP routines will also be on LOG along with any you generate.

Data Sets Required for LOG

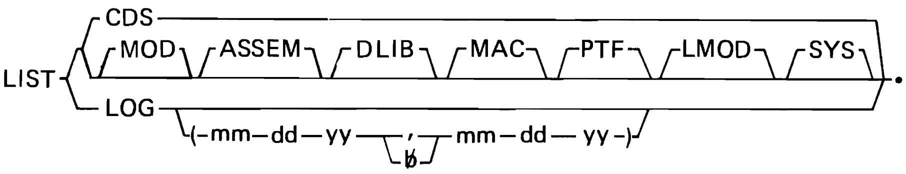
To use the LOG statement the following data sets are required:

SMPCNTL
SMPOUT
SMPLOG (LOG)
SMPCDS (CDS)

Detailed information about the data sets can be found in the write-up "Storage Requirements" in "Chapter 1: Introduction" and in the write-up "Data Sets" in this chapter.

LIST

The LIST statement will cause a print-out to be produced from either the CDS or the log data set (LOG). An example of the statements you can use to request a listing is provided earlier in this chapter under the topic "Other SMP Functions."



CDS

specifies all CDS members are to be printed. If the ACDS is to be listed, the SMPCDS DD statement must point to the ACDS.

option

specifies the type of CDS members to be printed. Any combination of the following can be coded:

MOD lists all distribution library modules that were chosen at system generation time, the DLIB data sets in which they are found, the module IDs, and the load modules to be updated on the target system when PTFs are applied to these modules. (A DLIB entry of 'SYSPUNCH' indicates that the module was assembled.)

ASSEM lists the assembler input for assembled modules.

MAC lists the cross-reference of the Stage II system generation macros and the modules that have been assembled. (Note that ++ MAC and ++ UPDTE modifications may change or create entries.)

PTF lists the PTFs on the system with the date they were processed. Also shows the status of the PTF, type of PTF (MOD, MAC, etc.), and the modules and macros that comprise the PTF.

LMOD lists all load modules in the target system that were linkedited or selectively copied at system generation, their link edit attributes, linkage editor control cards (if applicable), and the libraries to be updated with APPLY processing.

DLIB lists the libraries that were totally copied at system generation time and the library name on the target library to which they were copied.

SYS lists system information (such as the default NUCID and system release).

Note: PTF only should be specified if the ACDS is being listed.

LOG [(mm dd yy $\begin{Bmatrix} \not{b} \\ , \end{Bmatrix}$ mm dd yy)]

specifies that the contents of the LOG data set are to be printed. One date range may be specified as: mm the month, dd the day, and yy the year. Specify mm as 00 through 12, dd as 00 through 31, and yy as 00 through 99. Separate mm, dd, and yy with blanks.

Data Sets Required for LIST

To use the LIST statement the following data sets are required:

SMPOUT
SMPLOG (LOG)
SMPCDS (CDS) or SMPACDS (ACDS)
SMPCNTL

LIST LOG Messages

The following messages are produced by the LIST LOG function.

LIST REQUEST COMPLETED

Explanation: The requested listing is completed.

System Action: None.

Programmer Response: None.

END OF DATE. RANGE PROCESSING COMPLETED.

Explanation: The specified date range has been reached.

System Action: None.

Programmer Response: None.

THE DATE RANGE SPECIFIED IS INVALID.

Explanation: An incorrect date range has been decoded.

System Action: LIST processing is terminated.

Programmer Response: Correct the date range. Specify mm as 00 through 12, dd as 00 through 31, and yy as 00 through 99.

I/O ERROR. PROCESSING TERMINATED.

Explanation: An I/O error occurred when reading from the history log data set.

System Action: LIST processing is terminated.

Programmer Response: Retry the LIST function. If the I/O error is still encountered, then the history log data set is unusable. Proceed with your installation backup procedures.

END OF HISTORY LOG DATA SET.

Explanation: During LIST processing, the end of the data set has been reached.

System Action: None.

Programmer Response: None.

JCLIN

The JCLIN statement causes SMP to read in the Stage I output from system generation on similar data. This output is used as data input to update or create CDS. An example is provided earlier in this chapter under the topic "Other SMP Functions."

```
JCLIN ——————— .
```

For a complete re-generation (a system generation using the same DLIBs over again) the ACDS must be copied, using IEBCOPY, to the new CDS for the new system prior to JCLIN processing. This will transfer the PTF information for the DLIBs to the new system. This will maintain the PTF information about PTFs that have been put into the DLIBs.

For partial system generations, SMP should be run with the JCLIN statement specified. Module and macro entries on the CDS are updated. New assembler entries are stored with the new assembler input on the CDS. Linkage editor control statements for load module entries are replaced except for CHANGE and REPLACE information that is carried over to the updated version. Refer to the topic "Module Conflicts During Partial System Generation" in the Appendix for a description of procedures for a partial generation.

Data Sets Required for JCLIN

To use the JCLIN statement the following data sets are required:

```
SMPOUT            SMPCNTL
SMPLOG (LOG)      SMPJCLIN
SMPCDS (CDS)      SYSUT1
```

JCLIN Restrictions

The input for JCLIN processing must be a jobstream similar to an OS/VS system generation with the following restrictions:

- Each job and job step in the jobstream must be named and must end with /*
- Designate *assembly* steps as the cataloged procedure ASMS or as:

  ```
     PGM=IFOX00
     PGM=IEUASM
  or PGM=ASMBLR
  ```

  Indicate the member name (modname) of the assembled object module as:

  ```
     MOD=modname  (when using ASMS)
     //SYSPUNCH   DD     DSN=library(modname) , . . .
  or //SYSPUNCH   DD     DSNAME=library(modname) , . . .
  ```

- Designate *linkage editor* steps as the cataloged procedure LINKS or as PGM=IEWL.

  Indicate the data set name (libname) for the output of the linkage editor as:

  ```
     NAME=libname  (when using LINKS)
     //SYSLMOD     DD     DSN=index.libname, . . .
  or //SYSLMOD     DD     DSNAME=index.libname, . . .
  ```

- Execute IEBCOPY as: PGM=IEBCOPY

  Specify the COPY statement as:

  ```
  COPY    INDD=ddname1,OUTDD=ddname2
  ```

  Where ddname1 and ddname2 are equal to the lowest qualified data set name on the respective DD cards:

  ```
  COPY    OUTDD=LINKLIB,INDD=AOS12
  ```

Refer to the following topic "Adding New Modules" for an example of JCLIN. (See example 2).

UCLIN

The UCLIN statement is used to identify the statements immediately following it as update control statements (UCL) statements. An example is provided earlier in this chapter under the topic "Other SMP Functions."

```
UCLIN————————•
```

UCLIN

> specifies that UCL statements follow. They should be immediately after this statement.

Data Sets Required for UCLIN

To use the UCLIN statement the following data sets are required:

SMPOUT
SMPLOG (LOG)
SMPCDS (CDS) or SMPACDS (ACDS)
SMPCNTL

ENDUCL

The ENDUCL statement is used to identify the end of the update control language (UCL) statements. It must immediately follow the last UCL statement. An example is provided earlier in this chapter under the topic "Other SMP Functions."

```
ENDUCL———————•
```

ENDUCL

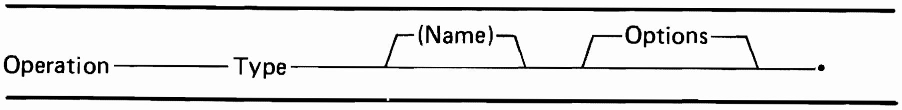
> specifies the end of the UCL statements.

Update Control Language (UCL)

The update control language (UCL) statements provide information that SMP uses to create, change, or delete entries in the CDS.

Control Statement Format – UCL Statements

The UCL control statements are written in a free-form style and have the following format:



The operation (ADD, REP, DEL), the type (PTF, MOD, LMOD, MAC, DLIB, SYS), and name are positional parameters. The options can be specified in any order.

Control Statements – UCL

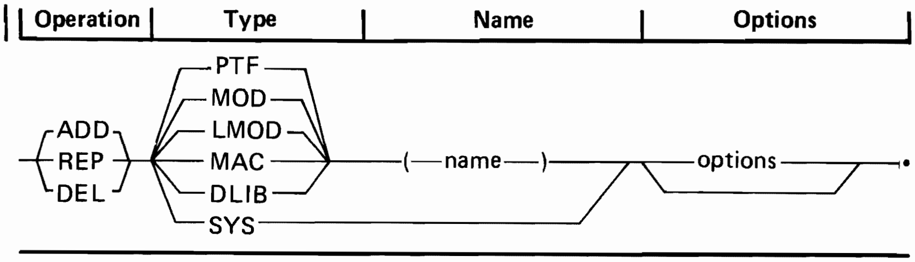
There are three operations performed by UCL control statements:

ADD – Will add data to an existing entry or create an entry in CDS or ACDS if none exists.

REP – Will cause an entry or fields within an entry in CDS or ACDS to be replaced by the information supplied in the keywords.

DEL – Will cause either an entire entry in CDS or ACDS to be deleted, if no keywords are specified, or the specified fields to be deleted.

For each control statement (ADD, REP, DEL) you must specify a 'type' plus a 'name' plus the options you selected. Each of these types with its name and options are described in Figure 11.



Only one type with name and various options can be specified in one statement. Following the statement format above and using the data from Figure 11 you will be able to construct the UCL statements you need.

Type	Name	Options	Descriptions
DLIB[1]	(library dsname)		1 to 8 characters — DLIB data set name
		SYSLIB(name[{b̸ ,} name] . . .)	ddname of DD statement for system library the DLIB was copied to
LMOD[2]	(load module name)		Operating system load module name 1 to 8 characters
		SYSLIB(name[{b̸ ,} name] . . .)	System library name, 1 to 8 characters
		AC=1	Authorization code (VS only)
		ALIGN2	Align on 2K boundary (VS only)
		COPYb̸	Copied at system generation; linkage editor attributes not available
		DCb̸	Downward-compatible
		NEb̸	Not editable
		OVLYb̸	Overlay attribute
		REFRb̸	Refreshable
		RENTb̸	Reentrant
		REUSb̸	Reusable
		SCTRb̸	Scatter loadable
		STDb̸	Standard linkage editor attributes only
MAC[1]	(macro name)		1 to 8 characters
		ASSEM(name[{b̸ ,} name] . . .)	Name of assembled module 1 to 8 characters
		ID(modification identifier)	2 characters
MOD[1]	(module name)[5]		1 to 8 characters — DLIB member name
		DLIB(library name)	DLIB name 1 to 8 characters
		ID(module identifier)	2 alphabetic characters
		LMOD(name[{b̸ ,} name] . . .)	Operating system load module name 1 to 8 characters
PTF	(ccnnnnn)		Modification number; cc modification identifier, nnnnn unique modification number
		ACPTb̸	Modification accepted
		APPLb̸	Modification applied
		REGEN	Modification in DLIB prior to system generation.
		DATE(yyddd)[3]	Date modification received
		MOD(name[{b̸ ,} name] . . .)[3,5]	DLIB member affected by modification
		MAC(name[{b̸ ,} name] . . .)[3]	Macro affected by modification
		ZAP(name[{b̸ ,} name] . . .)[3,5]	DLIB member affected by modification, which is a superzap.
		FORCEb̸	FORCE indicator
		SUP (ccnnnnn)[3]	Superceding modification number; cc modification identifier, nnnnn unique modification number.
		XZAP(name[{b̸ ,} name] . . .)[3,5]	Module affected by modification which is an EXPAND with superzap.

Figure 11. UCL Statements (Part 1 of 2)

Type	Name	Options	Descriptions
SYS			
		SREL(srrr)	s is system type – x=VS1, y=VS2 (SVS), z=VS2 (MVS), s=OS; rrr is release level i.e., 010 = release 1, 216 = release 21.6, etc.
		NUCID(default identifier)[4]	1 numeric character default nucleus identification used to save current nucleus; otherwise current nucleus not saved unless specified in APPLY statement..
		PEMAX (nnn)	Maximum number of PTF elements; default is 50.
		TSOƀ	Indicates TSO in the system.

[1] To create a new entry, all the keywords must be specified.

[2] ADD or REP cannot use this keyword to add or replace an entire load module type entry. Instead use the JCLIN control statement to obtain LMOD information from the Stage 1 output. ADD or REP can be used to change data in an existing load module entry.

[3] To create a new entry, DATE must be specified with SUP, or DATE must be specified with MOD, MAC, XZAP, or ZAP. (The SUP option must not be coded with MOD, MAC, XZAP, or ZAP.)

[4] This keyword is required when adding the system entry for the first time.

[5] Module name must be the name in the distribution library and not the load module name.

Figure 11. UCL Statements (Part 2 of 2)

Data Sets

SMP requires a variety of data sets. The total number is determined by the types of functions being executed. For the format of each data area, see Section 5, 'Data Areas' in *OS/VS System Modification Program (SMP) Logic.* A description that contains key information about each data set is provided below. The format used is:

Data Set	The common name of the data set is provided here.
Acronym	The characters commonly used for this data set are provided here.
ddname	The name supplied here must be used in the DD statement that is written for this data set.
Information	Information is supplied about the data set such as the contents, special information, and the type of structure used.
Device	The data supplied here identifies the type of device that can be used for the data set.

Data Set **Alternate Control Data Set**

Acronym ACDS

ddname SMPACDS

Information This data set (ACDS) contains the PTF information for the DLIBs. It should reside on one of the DLIB volumes and will contain data whenever PTFs have been put into the DLIBs – regardless of whether the PTF was put in before or after system generation.

This is a partitioned data set.

Device Direct access only.

Data Set **Control Data Set**

Acronym CDS

ddname SMPCDS

Information This data set (CDS) contains information about the macro instructions, modules, assemblies, load modules, libraries copied at system generation time, and PTFs in the system. The CDS is created from information collected from the Stage I output of system generation.

The CDS is updated by SMP routines or through the use of JCLIN or UCL. The data in the CDS is used by SMP to control the checking, inserting, or removing of modules and macro definitions. The ACDS data, if there is any, must be copied into the new CDS whenever the CDS is created.

Once the CDS has been created, you should run SMP after each partial system generation. This will update the CDS entries so that any PTFs you want to apply will be checked and processed correctly.

This is a partitioned data set.

Device Direct access only.

Data Set Link Libraries

ddnames distlib or tgtlib

Information These libraries contain the modules (distlib) and the load modules (tgtlib) that are defined in the CDS entries. The 'distlib' would be the ddname for a distribution library or permanent user library and 'tgtlib' would be the ddname for an operating system library.

These libraries are partitioned data sets.

When specifying ddnames for the target and distribution libraries, the name following SYS1. must be used as the ddname. For example:

SYS1.LINKLIB use LINKLIB as //LINKLIB DD

The ddnames used to define these libraries must be the lowest level index of the data set name, for example:

SYS1.LINKLIB has the ddname LINKLIB.

Device Direct access only.

Data Set History Log Data Set

Acronym LOG

ddname SMPLOG

Information This data set contains a time-stamped record of all significant events that occur during modification processing. SMP writes records automatically to this data set. The user can write records to LOG by using the LOG control statement. DISP=MOD should be specified so that a cumulative history is maintained.

This is a sequential data set. (It has RECFM=U, BLKSIZE=260 and should not be changed.)

The LIST control statement can be used to obtain a printout of all or selected portions of the information on the data set.

Device Tape or direct access.

Data Set **PTF Temporary Store**

Acronym PTS

ddname SMPPTS

Information Used as temporary storage for modules from SMPPTFIN, assembler output (object code) for macro instructions that are being updated, linkage editor EXPAND control statements, library names for modules stored in another library, IMASPZAP (or HMASPZAP, AMASPZAP) control cards, and IEBUPDTE control statements. The IEBUPDTE control cards are deleted after they are used by the RECEIVE function. The balance of the data is deleted during ACCEPT, REJECT, or RESTORE processing.

Do not use TSO EDIT to modify a member on the PTS, or directory entry information may be lost.

If a number of RECEIVE statements and APPLY statements are executed without any execution of ACCEPT, the space allocated for the PTS may be used up.

The space required by the PTS is determined by the number of PTFs that are in a RECEIVE or APPLY status. Therefore, PTFs in a RECEIVE status that are not going to be processed by APPLY or ACCEPT should be deleted by REJECT to free space. To regain the free space, compress the data set.

This is a partitioned data set.

Device Direct access only.

Data Set **Control Statement Input**

ddname SMPCNTL

Information Contains SMP control statements used to direct the execution of the SMP functions.

This is a sequential data set.

Device Card, tape, or direct access.

Data Set **Stage I Output From System Generation**

ddname SMPJCLIN

Information Contains the Stage I output from full or partial system generation (or other data in a similar format). (It should be the output from the most recent complete generation and any partial generations.)

This is a sequential data set.

Device Card, tape, or direct access.

Data Set **Macro Libraries (for SMP use)**

ddname SMPMACn

Information Each macro library used must be specified as a separate data set for SMP use as SMPMACn. These libraries are the same ones that are specified by the SYSLIB DD statement. Up to nine libraries can be defined. The n must be numeric and the numbers must be used sequentially – first '1,' then '2,' then '3,' and so on. During RECEIVE processing, SMP scans the SMPMACn DD cards until: (1) the macro is found in the library, (2) there is no SMPMACn for the n being searched, or (3) n=9 and the macro has not been found. (In cases 2 and 3, no macro updating occurs.)

These are the libraries that will be updated by the modifications. They should be at least the Stage I and II system generation macro libraries, and any other libraries, other than link libraries, that may be updated with a modification such as SYS1.PROCLIB.

If a number of RECEIVE statements are executed that process PTFs with macros without executing APPLY and ACCEPT, the space allocated for the macro libraries may be used up. Do a compress on the macro libraries from time to time to reclaim the unused space.

This is a partitioned data set.

Device Direct access only.

Data Set **Output Data Set**

ddname SMPOUT

Information Contains all SMP messages and listings.

This is a sequential.data set.

Device SYSOUT, printer, direct access, or tape.

Data Set **PTF or Modification Input**

ddname SMPPTFIN

Information Contains the PTFs or modifications to be put into the system. (See "Modification Control Statements Format" in this chapter for a detailed layout.)

This is a sequential data set.

Device Card, tape, or direct access.

Data Set **Macro Library (for assembler)**

Ddname SYSLIB

Information The macro library, concatenated if more than one, for use by the assembler. (Input only)

These libraries are the same macro libraries, concatenated in the same sequence, as those that are used for the Stage II of system generation.

This is a partitioned data set.

Device Direct access only.

Data Set	**Output Data Set**
ddname	SYSPRINT
Information	Contains all output generated by all invoked programs. This is a sequential data set.
Device	SYSOUT, printer, direct access, or tape.

Data Set	**Work Data Sets**
ddnames	SYSUT1, SYSUT2, SYSUT3
Information	Used as scratch data sets for SMP and any other programs called by SMP that require work data sets. These are sequential data sets.
Device	Direct access only.

Data Set	**Link and Text Libraries**
ddnames	lklib and txlib
Information	These libraries contain replacement modules or macros for use with the MOD and MAC modification control statements. The 'lklib' contains modules in link edited format. The 'txlib' contains modules in object format that have not been link edited or macros in source format. When LKLIB and TXLIB keywords are specified on the ++ MOD or ++ MAC statements, module or macro replacements do not follow the MOD or MAC modification statements. These libraries are partitioned data sets.
Device	Direct access.

Handling User Modifications

User modifications can be either changes to existing modules or adding new modules or load modules.

Changing Existing Modules

The kind of addition you are putting in the system will determine which modification control statements you will need. You will need a PTF header statement, possibly a VER statement, plus one or more of the other types of header statements. These statements are put into a data set defined as SMPPTFIN. A sample of the type of statements you might use is:

```
//STEP1 EXEC PGM=HMASMP
//SMPOUT DD SYSOUT=A
//SMPLOG DD DSN=SMPLOG,DISP=MOD
//SMPCDS DD DSN=SMPCDS,DISP=OLD
//LINKLIB DD DSN=SYS1.LINKLIB,DISP=OLD
//SMPPTS DD DSN=SMPPTS,DISP=OLD
//SYSPRINT DD SYSOUT=A
//SYSUT1 DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT2 DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SYSUT3 DD DISP=(,DELETE),UNIT=SYSDA,SPACE=(CYL,(2,1))
//SMPCNTL DD *
    RECEIVE  .
    APPLY S (MM00001) .
/*
//SMPPTFIN DD *
++ PTF (MM00001)  .
++ ZAP (IEFSD161)  .
    NAME IEFSD061
    VER B4 47F0C01E
    REP B4 47000000
/*
```

The distribution module name is specified on the ZAP statement. If the load module name is not specified on the superzap NAME card, then the superzap will be repeated in all load modules in which the module occurs. If the load module name is specified on the NAME card, processing is done only on that specific load module. (Note that the load module name must be valid for the module specified on the ++ ZAP card.) The NAME card must include the CSECT name. A complete description of all the modification control statements follows in the next topic.

Examples of Changing Existing Modules

The following examples show how to specify various applications for the PTF and ZAP modification control statements.

For the examples, assume that:

- LMODA contains DLIB modules – MOD1, MOD2, MOD3, MOD4.
- LMODB contains DLIB modules – MOD1, MOD4.
- MOD1 contains control sections – CSECT1, CSECT2, CSECT3.
- MOD2, MOD3, MOD4 have 1 CSECT each, CSECT name = module name

Example 1: To modify MOD1 CSECT specify:

```
//SMPPTFIN DD *
++ PTF(CS00001) .
++ ZAP(MOD1) .
   NAME   CSECT2
   VER    0D  FF
   REP    0D  FE
/*
```

In this example, at APPLY time, SMP will modify LMODA and LMODB. At ACCEPT time, SMP will modify MOD1.

Example 2: To modify MOD2 specify:

```
//SMPPTFIN DD *
++ PTF(CS00002) .
++ ZAP(MOD2) .
   NAME   MOD2
   VER    0A  00
   REP    0A  FF
/*
```

At APPLY time, SMP will modify LMODA only. At ACCEPT time SMP will modify MOD2.

Example 3: To modify only MOD1 CSECT1 in LMODB specify:

```
//SMPPTFIN DD *
++ PTF(CS00003) .
++ ZAP(MOD1) .
   NAME   LMODB CSECT1
   VER    00  00
   REP    00  FF
/*
```

At APPLY time, SMP will modify LMODB only. At ACCEPT time, SMP will modify MOD1 in the DLIBs.

Example 4: To modify MOD1 CSECT2 (with EXPAND) specify: (OS/VS only)

```
//SMPPTFIN DD *
++ PTF(CS00004) .
++ ZAP(MOD1) .
   EXPAND   CSECT2 (4)
   NAME     CSECT2
   VER      0D  FF
   REP      0D  FE
/*
```

At APPLY time, SMP will expand (by 4 bytes) and modify LMODA and LMODB. At ACCEPT time, SMP will expand (by 4 bytes) and modify MOD1.

Adding New Modules

When you are adding new modules or load modules to a system, an entry must first be created in the CDS so that SMP will recognize the modification as valid. The format of each type of CDS entry is provided in *OS/VS System Modification Program (SMP) Logic* under "Data Areas."

To add new modules to existing load modules you use the UCLIN statement followed by a UCL ADD statement. The ADD statement must contain the information required to construct the type of CDS entry required. Be sure you code the keywords that will supply the data SMP needs during the RECEIVE processing.

To add new load modules, create an SMPJCLIN data set containing a jobstream similar to a sysgen Stage II jobstream that would construct a load module from DLIB modules. Use this data set as SMPJCLIN input to the SMPJCLIN run.

Examples of Adding New Modules

The following examples show how to add a new module and a new load module to the system.

Example 1: Adding a New Module Entry to an Existing Load Module in the CDS

```
//UCLIN JOB ACCT,NAME,MSGLEVEL=1
//STEP1 EXEC PGM=HMASMP
//SMPOUT DD SYSOUT=A
//SMPLOG DD DSN=SMPLOG,DISP=MOD
//SMPCDS DD DSN=SMPCDS,DISP=OLD
//SMPCNTL DD *
   UCLIN .
   ADD MOD (IEFXYZ) /* ADDS AN ENTRY IN CDS FOR THIS MODULE */
         DLIB (AOSB3) /* NAME OF LIBRARY MODULE OCCURS IN */
         LMOD (IEANUC01) /* NAME OF SYSTEM LOAD MODULE OCCURS IN */ .
   ENDUCL .
/*
```

Example 2: Adding New Load Module and Module Entries in the CDS

```
//SGUPDAT  JOB   ACCT,NAME
//STEP1    EXEC PGM=HMASMP
//SMPOUT   DD   SYSOUT=A
//SMPLOG   DD   DSN=SMPLOG,DISP=MOD
//SMPCDS   DD   DSN=SMPCDS,DISP=OLD
//SYSUT1   DD   DSN=&SYSUT1,UNIT=SYSDA,SPACE=(CYL,(1,1))
//SYSPRINT DD   SYSOUT=A
//SMPCNTL  DD   *
     JCLIN .
/*
//SMPJCLIN DD   DATA,DLM='$$'
//PLN30    JOB  1,'SYS GEN',MSGLEVEL=1,CLASS=A
//STEP1    EXEC PGM=IEWL,PARM='RENT'
//SYSLMOD  DD   DSN=SYS1.LINKLIB,DISP=OLD
//USERLIB  DD   DSN=SYS1.USERLIB,DISP=OLD
//SYSLIN   DD   *
      INCLUDE   USERLIB(IEFUSERA)
      INCLUDE   USERLIB(IEFUSERB)
      ENTRY  USERONE
      NAME   USERTWO(R)
/*
//STEP2    EXEC PGM=IEBCOPY
//USERLIB  DD   DSN=SYS1.USERLIB,DISP=OLD
//SVCLIB   DD   DSN=SYS1.SVCLIB,DISP=OLD
//SYSIN    DD   *
      COPY    INDD=USERLIB,OUTDD=SVCLIB
      SELECT  MEMBER=(USERSVC1,USERSVC2)
/*
$$
```

In this example, load module entries for USERTWO, USERSVC1, and USERSVC2 are added to the LMOD section of the CDS. Module entries for IEFUSERA, IEFUSERB, USERSVC1, and USERSVC2 are added to the MOD section of the CDS.

Restoring the User ID

Whenever a re-sysgen is performed, that is doing a system generation from the same set of DLIBs as the previous sysgen, the modification ID on any modified modules or macro instructions must be restored. During the copying of the ACDS to the new CDS no IDs are carried over, so therefore any modification IDs must be replaced in order to maintain the integrity of the user modules or macro instructions. The UCLIN control statement is used followed by a UCL ADD statement.

See the following topic "PTF Header Statements" for a description of the user (modification) ID.
The following example shows a typical statement sequence that might be used.

```
//CDSUPDAT JOB ACCT,NAME,MSGLEVEL=1
//STEP1 EXEC PGM=HMASMP
//SMPOUT DD SYSOUT=A
//SMPLOG DD DSN=SMPLOG,UNIT=2314,VOL=SER=SMPVOL,DISP=MOD
//SMPCDS DD DSN=SMPCDS,UNIT=2314,VOL=SER=SMPVOL,DISP=OLD
//SMPCNTL DD *
   UCLIN .
   ADD MOD(IEFSD082) ID(MN) .
   ENDUCL .
/*
```

Modification Control Statements

The PTFs that are supplied by IBM are prepared in the format described in this section. Whenever you prepare modifications of your own to be included in the system by SMP, you must also use these statements.

The SMPPTFIN data set contains the modification control statements which define the PTFs and/or user modifications.

Note: The PTF header statement must be first and the VER statement (if it is coded) must be second. The other statements can appear in any sequence following the rules given for each in the detailed descriptions below. Remember that the statements will be processed in the sequence in which they occur.

Rules For Coding

The following rules should be followed when you are coding modification format control statements:

- Blanks may be used within each record.
- Each statement is terminated by a period (.).
- Each statement must start a new record or card.
- Comments are delineated by /* at the beginning and */ at the end. These comments can be used anywhere within the statement before the ending period but cannot begin in column 1, if the data set is on cards.
- Sequence numbers can be placed in positions (columns) 73 through 80 because these positions are ignored during SMP execution.
- The symbol '++' must be in columns 1 and 2 for modification control statements. The header identification such as PTF, VER, or MOD must be in the same record. For example, you cannot code '++' in one record and have PTF in the next record.

PTF Header Statement

This header statement indicates the beginning of a PTF or user modification and must be present. This statement must be the first one for each PTF or user modification.

```
++————PTF————(—ccnnnnn—)————•
```

cc

modification identification. Each IBM modification will start with 'U', 'A', or 'I'; these letters are restricted for IBM use only.

Note: This identification is placed in the module and macro instruction entries in CDS for checking purposes. When a PTF is verified by the RECEIVE statement, the identification must match with the CDS entry. If it does not the force indicator is set, and if you still want it in the system, you specify the FORCE keyword in the APPLY statement.

When you create your own modification, you should insert a two-character unique identifier in this field. This means that you become the owner of the module or macro definition. No modification can be made to your code unless it has the same two-character identifier. This ensures that only you can make changes or updates to this module or macro definition. After a unique identification is set by the ACCEPT statement, use the FORCE keyword (on the APPLY or ACCEPT statement) to process a PTF or modification with a different identifier.

Notes:

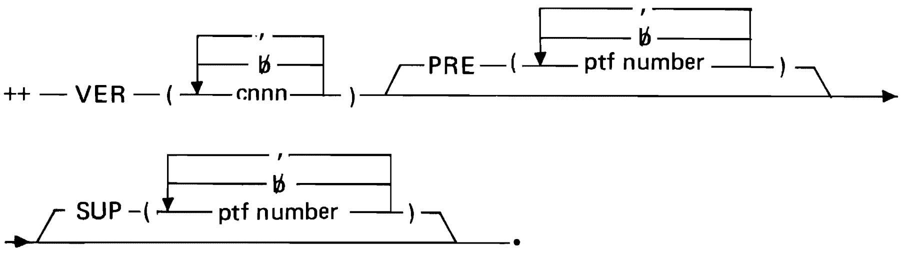
1. The ownership becomes in force only after the PTF has been processed by the ACCEPT statement.
2. The module and macro ID fields are initially blanks so no checking will be done for first PTF processed by RECEIVE.

nnnnn

is a five digit number, unique for each modification.

Verification Statement

This statement causes system and release level checking to be done. If this statement is not present, no checking is done. More than one verification statement can be in a PTF or user modification. Only one verification statement has to pass the checking process to allow the PTF or user modification to be put in the system.



c

is the system code; S = OS, X = VS1, Y = VS2 (SVS), and Z = VS2 (MVS).

nnn

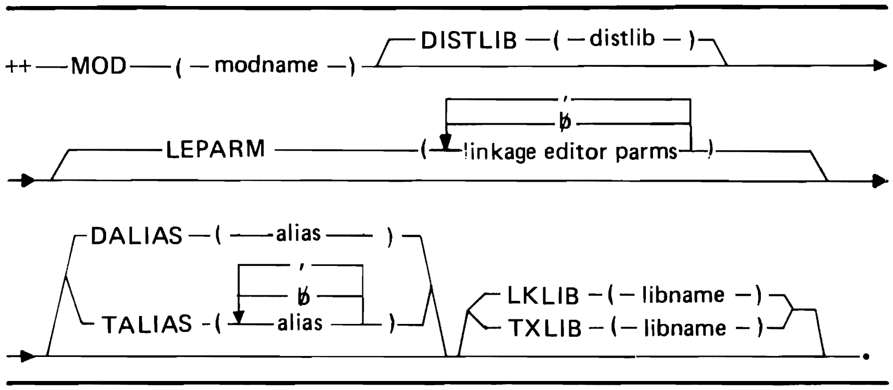
is the release number; for example, release 1 is coded as 010, release 21.6, is coded as 216, and so on.

PRE (ptf number)
specifies a prerequiste PTF or user modification number. That is the PTF or user modification specified must be in the system before the PTF or user modification being processed can be put in the system.

SUP (ptf number)
specifies a PTF or user modification that is superceded by the one being processed.

Module Header Statement

This statement describes one module that is in a PTF or modification. It must immediately precede the object code (if there is any to be entered in this input stream). If the module is in a partitioned data set that is a text or link library, the name of the library must be specified. In this case, the object code does not follow.



modname
is the distribution library module name.

DISTLIB (distlib)
is the distribution library ddname. This is only used if the module did not exist when the system generation was done or was part of a total data set copy at system generation. If the module is found on the CDS, then the DISTLIB keyword is ignored.

LEPARM (linkage editor parms)
any of the following linkage editor parameters can be specified for new modules:

RENT
REUS
SCTR
OVLY

REFR
DC
NE
STD

AC=1 (VS only)
ALIGN2 (VS only)

Specify STD to indicate that only the standard set (NCAL, XREF, LET, LIST) is to be used.

DALIAS (alias)
is an alias that exists only on a DLIB, there can be up to 5 aliases specified. This is used to describe modules that are included under an alias name during system generation.

TALIAS (alias)
is an alias that exists on an operating system library and a DLIB for copied modules; from 1 to 5 aliases may be specified.

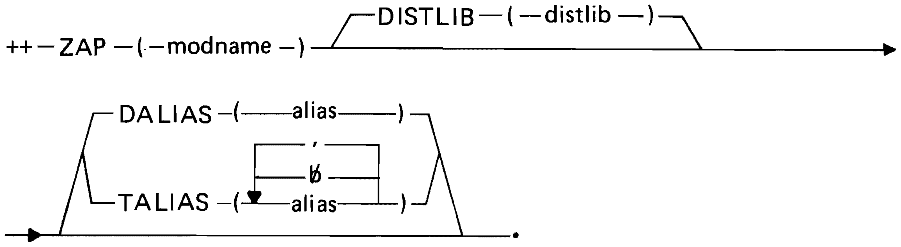
LKLIB (libname)
is the link library ddname. It contains replacement modules that are in link edited format.

TXLIB (libname)

is the text library ddname. It contains replacement modules that are in object format and have not yet been link edited.

SPZAP Header Statement

This statement describes SPZAP input. It must immediately precede the SPZAP control cards in the PTF or user modification.



modname

is the distribution library module name.

DISTLIB (distlib)

is the distribution library ddname. This is only used if the module did not exist when the system generation was done or was part of a total data set copy at system generation. If the module is found on the CDS, then the DISTLIB keyword is ignored.

DALIAS (alias)

is the alias that exists only on a DLIB. There can be up to five aliases specified. This is used to describe modules that are included under an alias name during system generation.

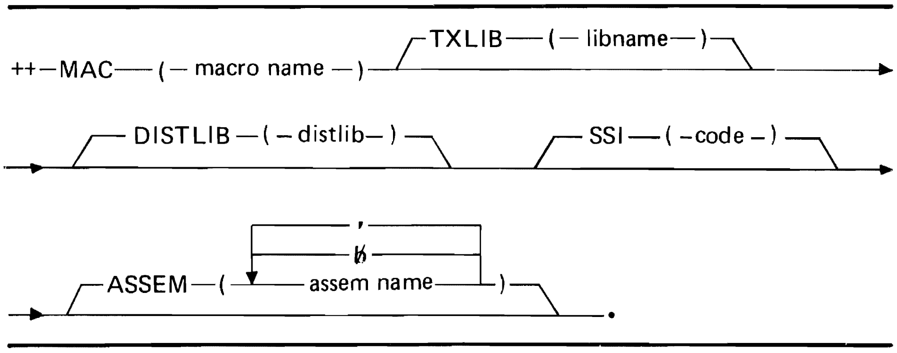
TALIAS (alias)

is an alias that exists on an operating system library and a DLIB for copied modules; from 1 to 5 aliases may be specified.

Note: When the NAME statement for SPZAP is coded with a single operand, it indicates that the superzap is to be applied to all the load modules in the system that contain this module. A name statement with a double operand, that is NAME OPERAND OPERAND indicates that the superzap is to be applied only to that load module.

Macro Header Statement

This statement describes one total macro replacement and must immediately precede the macro definition replacement if it is in this input stream. If it is an inner macro definition that does not appear during system generation, the module(s) to be reassembled must be specified in this statement. If the replacement macro definition is in a text library that is a partitioned data set and the replacement does not immediately follow, the text library name must be specified.



macroname

is the name of the macro instruction.

TXLIB (libname)

is the name of the text library where the replacement macro is located.

DISTLIB (distlib)

is the name of the lowest level index name. For example, SYS1.XYZ.TRS.ABC.MACLIBC would be coded as DISTLIB (MACLIBC).

When DISTLIB is specified, SMP does not scan sequentially for the macro but only scans SMPMACn which has the data set name referenced in DISTLIB. However, it is still important that you code the SMPMACn statements in ascending consecutive numerical order.

SSI (ccpfnnnn)

is the eight hexadecimal digits of system status information.

cc

is the change level of the module. Each release starts as 00. Each change increases it by one.

p

is X'1' indicating a PTF.

f

is flags.
X'1' IBM
X'5' Critical and IBM.
X'9' Dependency and IBM.

nnnn

are the last four digits of the PTF number.

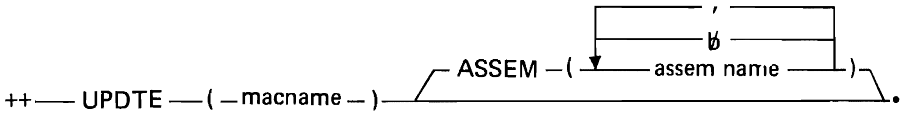
ASSEM (assem name)

is the name of the module(s) that have to be reassembled; a maximum of 50 can be specified.

Note: At RECEIVE time the module list from the ASSEM keyword is merged into the module list in the CDS macro entry. Thus any subsequent hit to that macro will cause reassembly of all modules of the original CDS entry plus any previous modules from the ASSEM keywords. The new ASSEM entries become a permanent part of the CDS.

IEBUPDTE Header Statement

This statement describes the IEBUPDTE control statements and must immediately precede the IEBUPDTE statements.



macname
: is the name of the macro instruction.

ASSEM (assem name)
: is the name of the module(s) that have to be reassembled; a maximum of 50 can be specified.

Note: At RECEIVE time the module list from the ASSEM keyword is merged into the module list in the CDS macro entry. Thus any subsequent hit to that macro will cause reassembly of all modules of the original CDS entry plus any previous modules from the ASSEM keywords. The new ASSEM entries become a permanent part of the CDS.

Note: The keyword UPDATE=INPLACE should not be in the IEBUPDTE CHANGE statement. If it is, the saved copy of the macro will be lost and this macro cannot be restored.

Program Temporary Fixes (PTFs)

IBM distributes its system modifications as program temporary fixes (PTFs) on a reel of magnetic tape. There will be one reel for each major operating system – OS, VS1, and VS2 – but the format of each tape will be identical. A cover letter will be sent with each tape and it will contain documentation of all the PTFs on that tape and the releases they apply to.

Sample PTF

The following example shows how a PTF might look. Each of the statements preceded by '++' is an SMP modification control statement.

This example shows a PTF that applies only to an OS system – the 'S' code – that supersedes a previous PTF and has a prerequisite PTF. There are two modules that are affected, IGC019AB and IEFSD105.

```
++ PTF  (US00005) .
++   VER   (S216)   PRE    (US00004)   SUP(US00003) .
++   MOD   (IGC019AB)   DISTLIB   (DM537)   .
     (object code occurs here)
++   MOD   (IEFSD105)   DISTLIB   (CI505)   .
     (object code occurs here)
```

There are some sample cataloged procedures provided in the appendix that can be used when putting PTFs in the system.

User Exit

An exit for a user-written routine has been provided during the RECEIVE processing. After each record is read from SMPPTFIN and before the record is processed, control is transferred to the user routine. The user routine must:

- Be link edited into the HMASMP load module.
- HMASMEXT must be the CSECT name.

Upon entry to HMASMEXT, register 1 points to the 80 byte buffer containing the record just read. The user can, by the use of a return code, have one or more records of his own read in and processed by RECEIVE. The return codes are:

Code	*Description*
0	Continue processing normally.
4	Return without a read the next time through the RECEIVE routine. This will allow the user routine to put a record in the buffer and have it processed by RECEIVE.
8	Delete this record.
12	Stop processing this PTF. Go to the next PTF and start processing that one. PTFs that are due to be deleted or bypassed will still be passed to the user exit routine.
16	Stop RECEIVE.
20	Stop HMASMP.

When an end-of-file condition occurs, the high order bit in register 1 will be turned on. If no more records are going to be added, put return code 0 in register 15. If more records are going to be added, put return code 4 in register 15.

Notes:

1. SMP does not clear the user exit buffer, therefore, the user exit routine should clear the buffer to prevent extraneous data from remaining in the buffer.
2. All PTFs are passed to the user exit whether or not they are processed by RECEIVE.

Appendix

This appendix contains the following topics that provide detailed information for the SMP user.

Messages

The messages on the following pages are put out at various times during execution of SMP. The following abbreviations have been used to indicate variable data within the message:

Abbreviation	Description
asn	Assembly name
ccc	Library entry name
ddn	ddname
eee	Element name
jjj	Job name
lib	Library name
mac	Macro instruction name
mod	Module name
nnn	Number
pgn	Program name
rc	Return code
sss	Step name
lll	Non-specific data
ppp	
xxx	
yyy	

HMA200I INVALID PARMS ON EXEC STATEMENT

Explanation: HMASMP has detected an error in the input parm list as stated on the JCL EXEC statement.

System Action: The HMASMP step is terminated. The message written to LOG is prefixed with the IPL date.

Programmer Response: Correct the PARM= JCL keyword and re-submit the job.

HMA201I UNABLE TO OPEN LIBRARY SPECIFIED BY ddn

Explanation: ddn is the ddname on a DD statement. HMASMP could not open a required library. Either a DD statement is missing or the referenced data set ddname specified by ddd is incorrect.

System Action: The action of HMASMP is indicated by the messages which follow.

Programmer Response: Add the DD statement required, correct the data set name, or perform other required steps to correct the problem.

HMA202I UNABLE TO OBTAIN STORAGE FOR WORK AREAS

Explanation: In attempting to GETMAIN storage for HMASMP internal tables, insufficient storage was available.

System Action: The HMASMP step is terminated at this point.

Programmer Response: Increase the REGION parameter on the EXEC statement (OS/VS2 or MVT) or increase partition size (OS/VS1 or MFT).

HMA203I SYNTAX ERROR IN $\left\{\begin{array}{l}\text{xxx CONTROL}\\ \text{yyy INPUT}\end{array}\right\}$ STATEMENT AT COLUMN nn

For PTF, VER, MOD, MAC, UPDATE, or ZAP:

Explanation: xxx is PTF, VER, MOD, MAC, UPDATE, ZAP, or UCL; nn is the column number; yyy is LINKAGE EDITOR, ASSEMBLER, or IEBCOPY. During the RECEIVE operation, one or more keywords associated with this PTF file record is in error at column nn.

System Action: If error is with the PTF or VER header, the PTF data will be flushed until the next PTF header is found, if any.

Programmer Response: Check the format of the keywords associated with this record and correct any errors in that statement on the PTF file.

For UCL:

Explanation: nn is the column number. A syntax error was detected in the UCL control statement at column 'nn'.

System Action: The update request is ignored. Processing continues with the next UCL control statement.

Programmer Response: Correct the UCL control statement and re-run the update to the SMP CDS library.

For LINKAGE EDITOR, ASSEMBLER, or IEBCOPY:

Explanation: A syntax error occurred during the JCL and control card scan for JCLIN.

System Action: The scan is terminated at this point.

Programmer Response: Correct the error and re-submit the job.

HMA204I ERROR FORCES HMASMP TO TERMINATE

Explanation: An error occurred, the details of which are described in a previous message, that forces HMASMP to terminate processing at this point.

System Action: The HMASMP processing is terminated.

Programmer Response: Correct the error indicated in the previous message and re-submit.

HMA205I HMASMP PROCESSING COMPLETED – HIGHEST RETURN CODE IS rc

Explanation: rc is the return code.

System Action: The return code specified by rc is passed to the system.

Programmer Response: None.

HMA206I USER EXIT RETURN CODE INDICATES TERMINATION OF $\left\{\begin{array}{l}\text{PTF}\\ \text{RECEIVE}\\ \text{HMASMP}\end{array}\right\}$

Explanation: The return code from the user exit routine after reading a record from the PTF data set requests the termination.

System Action: The user-requested processing will be terminated.

Programmer Response: If termination was erroneous, check the user exit routine to insure that the correct return code is passed back.

HMA207I UNKNOWN USER EXIT RETURN CODE – RECEIVE TERMINATED

Explanation: The return code from the user exit routine after reading a record from the PTF file is undefined.

System Action: The RECEIVE function is terminated.

Programmer Response: Check the logic of the user exit routine to insure that only defined codes are returned to the RECEIVE function.

HMA208I RECEIVE PROCESSING COMPLETED

Explanation: Normal termination for the RECEIVE function.

System Action: Processing continues.

Programmer Response: None.

HMA209I RECEIVE PROCESSING TERMINATED

Explanation: An error has occurred which warrants termination of the RECEIVE operation.

System Action: Reading of the PTF file is terminated. There may be more PTFs on the tape.

Programmer Response: A prior message has described the reason for the error or no SMPMACn DD card reflected a data set which contained a macro to be updated or replaced. Correct the error and resubmit the job.

HMA210I PTF nnn ALREADY RECEIVED

Explanation: nnn is the PTF number. The PTF being processed has previously been processed by a RECEIVE control statement.

System Action: Processing of the PTF will continue only if this PTF has been specified with a SELECT keyword.

Programmer Response: If the current PTF was not specified with the SELECT keyword, but you want it to replace the previously received PTF, execute SMP again and specify the SELECT keyword for this PTF.

HMA211I FOLLOWING PREREQUISITE PTF(S) NOT APPLIED AND ACCEPTED

Explanation: The listed prerequisite PTF numbers have not been applied and accepted.

System Action: The force-indicator is set in the SMPCDS PTF entry and processing will continue.

Programmer Response: Review the list of PTF numbers following this message to determine why they were not processed by SMP. If you want to process the prerequisite PTFs with a RECEIVE control statement, reject this PTF then issue RECEIVE, APPLY, and ACCEPT control statements for the prerequisite PTFs.

HMA212I nnn $\left\{\begin{matrix}\text{RECEIVED}\\ \text{APPLIED}\\ \text{ACCEPTED}\end{matrix}\right\}$ BUT NOT $\left\{\begin{matrix}\text{APPLIED}\\ \text{ACCEPTED}\\ \text{APPLIED}\end{matrix}\right\}$
NOT RECEIVED

Explanation: nnn is the PTF number. This message identifies the prerequisite PTFs that have not been processed by an APPLY or ACCEPT statement.

System Action: None.

Programmer Response: Scan through the prerequisite PTFs to ensure that no conflicts will result.

HMA213I PTF ID (xx) DIFFERS FROM MODULE mod ID (zz)

Explanation: xx is the PTF ID; mod is the module name, and zz is the module ID. The PTF identifier differs trom the identifier associated with the named module.

System Action: Processing of the PTF will continue, the force indicator is set, and validity checking continues.

Programmer Response: Determine the reason why the module identifier is different. If due to user modifications, insure that this PTF is compatible with the current version of the module or macro before selecting this PTF.

HMA214I STORE FAILED FOR $\left\{\begin{matrix}\text{MACRO mac}\\ \text{MODULE mod}\end{matrix}\right\}$ ON lib LIBRARY

Explanation: mac is the macro name; mod is the module name and lib is the library ddname. The directory entry for this macro or module cannot be stored.

System Action: Processing for this PTF containing this macro or module is terminated.

Programmer Response: Check the directory of the system library in which this macro's directory entry was to be stored before trying to issue the RECEIVE control statement for this PTF again.

HMA215I $\left\{\begin{matrix}\text{mac}\\ \text{mod}\end{matrix}\right\}$ $\left\{\begin{matrix}\text{FOUND}\\ \text{NOT FOUND}\end{matrix}\right\}$ ON yyy LIBRARY $\left\{\begin{matrix}\text{(ASSEMBLY MODULE)}\\ \text{(BACKUP COPY)}\end{matrix}\right\}$

Explanation: mod is the module name; mac is the macro name; yyy is SMPPTS or a macro library name. The named module or macro was either FOUND ON the specified LIBRARY or NOT FOUND ON SMP CDS LIBRARY. If '(REASSEMBLY MODULE)' appears then the named module or macro was to be reassembled because of the macro being processed by a RECEIVE statement. If '(BACKUP COPY)' appears then the named macro had a backup copy (probably from a previously received PTF) or its backup copy could not be found.

System Action: If the named module or macro was found on the specified library processing for this PTF is terminated. If it was not found on SMPCDS library, the force bit is set and processing continues.

Programmer Response: If the module or macro was found, the probable cause is that a PTF that was previously processed by a RECEIVE or APPLY statement caused this situation. Issue an ACCEPT, REJECT, OR RESTORE statement for that previous PTF before issuing a RECEIVE statement for this one again. If it was found when the PTF is processed by a RECEIVE statement, the force indicator will be set. This PTF must then be processed with an ACCEPT statement with the FORCE keyword specified.

HMA216I IEBUPDTE FAILED FOR mmm – RETURN CODE = rc

Explanation: mmm is the macro name, rc is the return code. IEBUPDTE failed to complete successfully for the named macro. The code rc was passed from the IEBUPDTE program.

System Action: Processing for this PTF is terminated.

Programmer Response: Determine the reason why IEBUPDTE failed by referring to the Utilities Message SRL. Correct the error and rerun the job.

HMA217I RESTORING mac ON lib LIBRARY

Explanation: mac is the macro name, lib is the library name. Processing for this PTF could not be completed successfully.

System Action: The named macro is being restored on the specified library.

Programmer Response: This message will be followed with message HMA218I when restoring is complete. This will indicate that the macro is restored to its pre-received state.

HMA218I SUCCESSFULLY STORED zzz xxx ON lib LIBRARY

Explanation: zzz is MODULE, LOAD MODULE, MACRO, PTF, ASSEMBLY, DISTLIB, NUCLEUS, UPDTE, ZAP, MEMBER, SYSTEM ENTRY; xxx is the name of a module, load module, macro, PTF, assembly, distlib, nucleus, UPDTE, ZAP, member, or system entry; lib is the library name. The named module, load module, macro, PTF, assembly, DISTLIB, nucleus, UPDTE, zap, member, or system entry was successfully stored or restored.

System Action: Processing continues.

Programmer Response: None.

HMA219I PTF ELEMENT MAXIMUM EXCEEDED

Explanation: The SMPCDS library entry cannot be created, updated, or listed because the PTF, module, or macro requires a PEMAX value greater than presently specified in the SMPCDS system entry.

System Action: The requested function is terminated.

Programmer Response: Increase the SMPCDS system entry PEMAX value using the UCLIN function. The PEMAX value should not be decreased after PTFs have been processed with a specific PEMAX value.

HMA220I MAXIMUM I/O ERRORS EXCEEDED

Explanation: A limit of I/O errors on the PTF file was exceeded.

System Action: The RECEIVE function is terminated.

Programmer Response: If the cause of the I/O errors is hardware, read the PTF file from another device.

HMA221I PTF nnn PROCESSING TERMINATED

Explanation: nnn is the PTF number. Error(s) processing this PTF are serious enough to cause its termination.

System Action: This PTF is terminated. Restoring of the SMPPTS library and macro libraries, if any, begins.

Programmer Response: Analyze previous error messages to determine if the RECEIVE statement for this PTF can be issued again by either correcting the previous errors or by specifying SELECT in the RECEIVE statement. Note: By using the SELECT facility of RECEIVE, certain error conditions are overridden.

HMA222I PTF nnn RECEIVED – FORCE INDICATOR SET – FOLLOWING SYSTEM LIBRARIES WILL BE UPDATED BY APPLY

Explanation: nnn is the PTF number. The RECEIVE statement for this PTF has been successfully executed. The following list of system libraries will be required if APPLY or RESTORE statements are issued for this PTF.

System Action: Processing of the next PTF (if any) begins.

Programmer Response: None.

HMA223I lib

Explanation: lib is the library name. This message follows the message number HMA222I and identifies the libraries that will be updated.

System Action: None.

Programmer Response: None.

HMA224I SUCCESSFULLY DELETED zzz $\left\{\begin{matrix}\text{mac}\\ \text{mod}\end{matrix}\right\}$ FROM THE lib LIBRARY

Explanation: zzz is MODULE, LOAD MODULE, MACRO, PTF, DISTLIB, NUCLEUS, MEMBER, SYSTEM ENTRY, UPDTE, ZAP, or ASSEMBLY; mod is a module name; mac is the macro name; lib is a library name. The named module or macro was successfully deleted from the named library.

The reasons are:

- Modules deleted from the PTS due to a REJECT, RESTORE, or ACCEPT operation.
- Macros deleted from the SMPMACn library due to:
  1. A RESTORE or REJECT operation.
  2. During RECEIVE, a new macro is deleted due to a previous error (such as an assembly or I/O error).
  3. During ACCEPT, an old macro back-up copy is deleted.

System Action: Processing continues.

Programmer Response: None.

HMA225I PTF nnn NOT APPLICABLE TO THIS SYSTEM – yyy

Explanation: nnn is the PTF number; yyy is the current system and release. The PTF selected but no system or release value found on the verify statement matched the current system and release value.

System Action: If SELECT was specified, then the FORCE bit is set. If SELECT was not specified, then the PTF is flushed.

Programmer Response: If this PTF should not have been selected because it is not applicable to this system, then issue a REJECT statement for it. If it is wanted in the system, issue an APPLY statement with the FORCE keyword specified.

HMA226I xxx PROCESSING TERMINATED FOR PTF nnn DUE TO ERROR

Explanation: xxx is ACCEPT, APPLY, REJECT, or RESTORE; nnn is the PTF number. Processing has been terminated for the PTF due to an error.

System Action: The PTF in error is bypassed and processing continues.

Programmer Response: Check previous messages to determine the cause of error. Correct the error and re-run the job.

HMA227I xxx PROCESSING SUCCESSFULLY COMPLETED FOR PTF nnn

Explanation: xxx is ACCEPT, APPLY, REJECT, or RESTORE; nnn is the PTF number. Processing has been successfully completed for the specified function.

System Action: None.

Programmer Response: None.

HMA228I IEANUC01 NOT FOUND ON NUCLEUS LIBRARY

Explanation: The nucleus, IEANUC01, was not found on the nucleus library as a result of a BLDL operation.

System Action: Processing is terminated for this function.

Programmer Response: Create IEANUC01 or specify a different NUCLEUS DD card.

HMA229I CONTROL STATEMENT IGNORED DUE TO PREVIOUS ERROR

Explanation: Previous error caused this control statement to be bypassed.

System Action: Processing continues with next statement.

Programmer Response: Correct the cause of the previous error and rerun this job.

HMA230I IEHIOSUP EXECUTED FOR $\left\{\begin{matrix}\text{APPLY}\\ \text{RESTORE}\end{matrix}\right\}$ –RETURN CODE = rc

Explanation: rc is the return code. The IEHIOSUP program was executed to update the TTR entries in the transfer control tables of the SVCLIB.

System Action: Processing for this statement is terminated if the return code is non-zero.

Programmer Response: None.

HMA231I IMASPZAP CONTROL STATEMENT ERROR IN MODULE mod FOR PTF nnn

Explanation: mod is module name; nnn is PTF number. A syntax error occurred in the control statement for IMASPZAP in the named module.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: Correct the syntax error and re-run the HMASMP step.

HMA232I ASSEMBLY asn FOR PTF nnn NOT FOUND ON SMPCDS LIBRARY

Explanation: During the RECEIVE processing of PTF number nnn of a macro which requires reassemblies to be performed, the assembly asn was not found on the CDS.

System Action: The force indicator is set for this PTF.

Programmer Response: This message usually occurs while RECEIVE is processing PTFs with a NULL CDS. If the CDS exists, then this could be a nonapplicable PTF.

HMA233I $\left\{\begin{matrix}\text{MODULE}\\ \text{MACRO}\\ \text{PTF}\end{matrix}\right\}$ xxx NOT DELETED FROM $\left\{\begin{matrix}\text{SMPPTS}\\ \text{MACLIB}\\ \text{SMPCDS}\end{matrix}\right\}$ LIBRARY FOR PTF nnn DUE TO ERROR

Explanation: xxx is the module name, macro name, or PTF number; nnn is the PTF number. The module, macro, or PTF was not deleted from the SMPPTS, macro library, or SMPCDS library or a macro could not be deleted from the macro library specified by SMPMACn due to an error for this PTF.

System Action: Processing for this PTF is terminated.

Programmer Response: None.

HMA234I BLDL FAILED FOR PROGRAM pgm REQUIRED FOR HMASMP EXECUTION

Explanation: pgm is the program name. The program indicated by pgm is required in order for HMASMP to execute.

System Action: The step is terminated.

Programmer Response: Add the program indicated to the JOBLIB, STEPLIB, or link library. This problem could occur due to an invalid name specified on the HMASMP EXEC statement PARM= parameter. In this case, correct the name and resubmit the job.

HMA235I PTF nnn NOT FOUND ON SMPCDS LIBRARY

Explanation: nnn is the PTF number. The indicated PTF is not found in the CDS and has been selected for APPLY, ACCEPT, REJECT, or RESTORE.

System Action: This PTF is ignored.

Programmer Response: Select the proper PTF and resubmit the job.

HMA236I $\left\{\begin{matrix}\text{ORIGINAL}\\ \text{UPDATED}\end{matrix}\right\}$ MACRO mac NOT FOUND ON MACLIB FOR PTF nnn DUE TO ERROR

Explanation: mac is the macro name; nnn is the PTF number. The updated macro could not be found for the ACCEPT function and this resulted in not deleting the original macro from the macro library. Or the original macro could not be found for the RESTORE or REJECT function and that results in not deleting the updated macro from the macro library.

System Action: Processing for this PTF is terminated.

Programmer Response: Determine and correct cause of error and rerun job.

HMA237I SMPCDS LIBRARY xxx ENTRY yyy UPDATE FAILED FOR PTF nnn DUE TO ERROR

Explanation: xxx is PTF, MACRO, or MODULE; yyy is the PTF number, the macro name, or the module name; nnn is the PTF number. An error occurred updating (STOW REPLACE) a PTF, MACRO, or MODULE entry in the SMPCDS library.

System Action: Processing is terminated for this PTF.

Programmer Response: None.

HMA238I $\left\{\begin{matrix}\text{LINK}\\ \text{COPY}\end{matrix}\right\}$ PROCESSING COMPLETED FOR $\left\{\begin{matrix}\text{LOAD MODULE}\\ \text{MODULE}\end{matrix}\right\}$ mod IN PTF nnn – RETURN CODE = rc

Explanation: mod is the module name; nnn is the PTF number; rc is the return code from the linkage editor or IEBCOPY. An execution of IEBCOPY or the linkage editor returned a code of 8 or less. Since multiple PTFs may be linked or copied, some PTFs may have been successfully applied, even though an error code was returned. This message indicates that all modules and load modules handled during this run have an error, although only one may have failed. Also, this message may appear for modules within a PTF that were never linkedited if other modules in the PTF have had linkedits that failed.

System Action: APPLY – PTF entries marked applied.
ACCEPT – PTF entries marked accepted.
RESTORE – PTF entries restored.

Programmer Response: Check the output from the linkage editor or IEBCOPY to identify the status of each PTF. If the PTF mentioned in the message was successful, no action is required for APPLY and ACCEPT.

HMA239I $\left\{\begin{matrix}\text{LINK}\\\text{COPY}\end{matrix}\right\}$ **FAILED FOR** $\left\{\begin{matrix}\text{LOAD MODULE}\\\text{MODULE}\end{matrix}\right\}$ **mod IN PTF nnn – RETURN CODE = rc**

Explanation: mod is the module name; nnn is the PTF number; rc is the return code from the linkage editor or IEBCOPY. An execution of IEBCOPY or linkage editor returned a code greater than 8. Since multiple PTFs may be linked or copied, some PTFs may have been successfully applied, even though an error code was returned. This message indicates that all the modules and the load modules handled during this run have an error, although only one may have failed. Also, the message may appear for modules within a PTF that were never linkedited if other modules in the PTF have had linkedits that failed.

System Action: APPLY – PTF entries marked applied.
ACCEPT – PTF entries marked ACCEPTED.
RESTORE – PTF entries unchanged.

Programmer Response: Check output from the linkage editor or IEBCOPY to identify status of each PTF. If the PTF mentioned in the message was actually successful, no action is required for APPLY and ACCEPT. For RESTORE, use the UCLIN to delete the PTF entry.

If the PTF is in error, correct the error and resubmit the job.

HMA240I ASSEMBLY COMPLETED FOR MODULE mod RETURN CODE = rc

Explanation: mod is the module name; rc is the return code. Processing has been completed for the module specified. If the return code equals zero, then SMP continues processing. If the return code is between zero and 16, then SMP sets the FORCE indicator and continues processing. If the return code is 16 or greater, then SMP terminates processing and backs off whatever has been done to this point.

System Action: None.

Programmer Response: Check assembly listing if return code is non zero.

HMA241I SELECTED PTF nnn $\left\{\begin{matrix}\text{NOT FOUND ON lib LIBRARY}\\\text{IS SUPERCEDED}\end{matrix}\right\}$

Explanation: nnn is the PTF number; lib is the library name. The selected PTF could not be found on the specified library or is a dummy PTF entered into the SMPCDS library because of a superceeding PTF.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: Probable user error. Check PTF numbers for accuracy. Use superceeding PTF.

HMA242I PTF nnn SELECTED FOR ACCEPT $\left\{\begin{matrix}\text{HAS NOT BEEN APPLIED}\\\text{HAS BEEN ACCEPTED}\end{matrix}\right\}$

Explanation: nnn is the PTF number. Before an ACCEPT statement can be issued for a PTF, an APPLY statement must be issued. This can be overridden by use of the force option.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: For HAS NOT BEEN APPLIED – Code the FORCE keyword on the ACCEPT statement or use the APPLY statement to process the PTF. For HAS BEEN ACCEPTED – Code the FORCE keyword on the ACCEPT statement.

HMA243I PTF nnn SELECTED FOR APPLY $\left\{\begin{matrix}\text{HAS BEEN ACCEPTED}\\\text{HAS FORCE INDICATOR SET}\end{matrix}\right\}$

Explanation: nnn is the PTF number. This PTF is in an accepted state or has already been applied or requires the FORCE keyword.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: Probable user error. For HAS BEEN ACCEPTED – You must issue a RECEIVE statement using SELECT for the PTF and then issue an APPLY statement to put the PTF in the system. For HAS BEEN APPLIED or HAS FORCE INDICATOR SET – Code the FORCE keyword on the APPLY statement to get this PTF into the system.

HMA244I PTF nnn SELECTED FOR REJECT HAS BEEN $\left\{\begin{matrix}\text{ACCEPTED}\\\text{APPLIED}\end{matrix}\right\}$

Explanation: nnn is the PTF number. A REJECT statement cannot be issued for a PTF that has been processed by an APPLY or ACCEPT statement.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: If the PTF has been processed by an APPLY statement, a RESTORE statement must be issued to remove it from the system. If an ACCEPT statement has been issued for this PTF, a REJECT statement cannot be used.

HMA245I PTF nnn SELECTED FOR RESTORE $\left\{\begin{matrix}\text{HAS NOT BEEN APPLIED}\\\text{HAS BEEN ACCEPTED}\end{matrix}\right\}$

Explanation: nnn is the PTF number. For 'APPLIED', a PTF must have been processed by an APPLY statement before a RESTORE statement can be effective. For 'ACCEPTED', a RESTORE statement cannot be issued for a PTF once it has been processed using an ACCEPT statement.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: Determine from SMPCDS library listing the status of the PTF. If an APPLY statement has not been issued, a REJECT statement can be issued. If it has not been processed using an ACCEPT statement, a RESTORE statement can be issued for this PTF.

HMA246I MODULE mod for PTF nnn NOT FOUND ON lib LIBRARY

Explanation: mod is the module name; nnn is the PTF number; lib is the library name.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF for this SMP statement. Any other statement will be flushed with message HMA229I.

Programmer Response: Probable user error. Determine from the SMPLOG the reason for this module missing from the SMPPTS library. Determine from the system generation listing the reason for the missing module. Probably the PTF does not apply to the system.

HMA247I BLDL FAILED IN LIBRARY lib FOR LOAD MODULE mod IN PTF nnn

Explanation: mod is the load module name; nnn is the PTF number. The BLDL issued to obtain linkage editor parameters failed for this load module.

System Action: When an APPLY statement is processing this PTF, the PTF is terminated. When an ACCEPT statement is processing this PTF, then a default set of LEPARMS is used; RENT, REUS, DC, and REFR.

Programmer Response: When this PTF is being processed by an APPLY statement, it is a probable user error. Either the library name is incorrect or the DD statement is missing. When an ACCEPT statement is being processed, there is no programmer response.

HMA248I THE xxx OPTION WAS REQUESTED – NO PTFS MEET SPECIFICATIONS

Explanation: xxx is APPLY, ACCEPT, REJECT, or RESTORE. The requested option is rejected because no PTFs meet the required specifications.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF for this SMP statement. Any other statement will be flushed with message HMA229I.

Programmer Response: Review other messages issued during this run and check the Processing to determine the problem.

HMA249I PTF nnn REJECTED BECAUSE OF NAME CARD CONFLICT IN MODULE mod

Explanation: nnn is the PTF number; mod is the module name. It is not permissible to have name cards of different types within the same superzap; (NAME OPERAND) and (NAME OPERAND OPERAND).

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: If this is a system PTF, call IBM for support. If it is a local modification, correct the NAME cards and re-run job.

HMA250I PTF nnn HAD A VERIFY REJECT IN MODULE mod

Explanation: nnn is the PTF number; mod is the module name.

System Action: Processing of the named PTF is terminated. Processing will continue with the next PTF.

Programmer Response: Check IMASPZAP dump of module for reason of failure.

HMA251I PTF nnn – A RETURN CODE GREATER THAN ZERO RETURNED FROM THE REP FUNCTION OF IMASPZAP

Explanation: nnn is the PTF number.

System Action: Processing will continue with this PTF.

Programmer Response: Check IMASPZAP listing of modules for this PTF on SYSOUT.

HMA252I INCOMPLETE HMASMP CONTROL STATEMENT

Explanation: HMASMP detected an incomplete control statement. An end-of-file occurred before the end of the statement.

System Action: HMASMP processing is terminated. The HMASMP function specified is not performed.

Programmer Response: Check for missing comment terminator (*/) or missing statement terminator (.) or a previous LOG statement with unbalanced parentheses.

HMA253I ENTRY ccc TO BE $\left\{\begin{array}{l}\textbf{REPLACED}\\ \textbf{DELETED}\\ \textbf{ADDED}\end{array}\right\}\left\{\begin{array}{l}\textbf{DOES NOT}\\ \textbf{ALREADY}\end{array}\right\}$ **EXISTz ON lib LIBRARY**

Explanation: ccc is the SMPCDS library entry name; z is a blank or S· lib is the library name. During an update of the SMPCDS library it was found that the entry currently exists and was to be added or the entry was to be changed, deleted, or replaced and could not be located.

System Action: The update request is ignored. Processing continues with the next UCL control statement.

Programmer Response: Correct the UCL control statement and re-run the update to the SMPCDS library.

HMA254I WARNING – PTF ELEMENT MAXIMUM REDUCED TO xxx

Explanation: xxx is the new PTF element maximum (PEMAX) value. Update to the system entry caused the PTF element limit (PEMAX) to be reduced. This may cause existing SMPCDS PTF entries to be too large for HMASMP processing.

System Action: The system PEMAX entry is updated to the new PEMAX value.

Programmer Response: None.

HMA255I UPDATE COMPLETE FOR ccc

Explanation: ccc is the SMPCDS library entry name. Update processing for the SMPCDS library entry was completed.

System Action: Update processing continues with the next UCL control statement.

Programmer Response: None.

HMA256I UPDATE PROCESSING TERMINATED – UPDATE NOT COMPLETE FOR ccc

Explanation: ccc is the SMPCDS library entry name. Update processing for the SMPCDS library entry was not completed due to an I/O error.

System Action: Update processing is terminated.

Programmer Response: Correct the source of the I/O error (as described by a previous error message and re-run the updating of the SMPCDS library.

HMA257I INSUFFICIENT DATA SUPPLIED FOR UPDATE TO ccc

Explanation: ccc is the SMPCDS library entry name. Insufficient data was supplied to perform the SMPCDS library update.

System Action: The update request is ignored. Processing continues with the next UCL control statement.

Programmer Response: Provide the missing information and re-run the update to the SMPCDS library.

HMA258I END OF FILE ON UCL INPUT STREAM – PROCESSING TERMINATED

Explanation: End of file occurred on the input data set before the 'ENDUCL' control statement was found.

System Action: The current UCL control statement is ignored and UCL processing is terminated.

Programmer Response: Correct the input data stream and include the 'ENDUCL' control card. Rerun the update to the SMPCDS library.

HMA259I MAXIMUM NUMBER OF LIST ELEMENTS EXCEEDED FOR ccc (eee)

Explanation: ccc is the SMPCDS library entry name; eee is the list element. A request to add a list element to SMPCDS library entry will cause the list length to exceed the maximum allowed.

System Action: The update request is ignored. Processing continues with the next UCL control statement.

Programmer Response: Reduce the number of list entries in the update control statement or increase the PEMAX value in the system entry.

HMA260I LIST ELEMENT eee $\left\{\begin{array}{l}\text{NOT FOUND}\\ \text{ALREADY EXISTS}\end{array}\right\}$ **FOR ccc**

Explanation: eee is the list element name; ccc is the SMPCDS library entry name. User requested a delete of list element for the SMPCDS library entry and it could not be found; or requested a list element to be added and the element currently exists in the SMPCDS library entry.

System Action: The update request is ignored. Processing continues with the next UCL control statement.

Programmer Response: Correct the UCL control statement and re-run the update to the SMPCDS library.

HMA261I STORE FAILED FOR ALTERNATE CDS ENTRY FOR PTF nnn IN SMPACDS LIBRARY

Explanation: nnn is the PTF number. During an ACCEPT operation, the PTF entry specified by nnn could not be stored in the SMPACDS library due to an error explained in a prior message.

System Action: The processing of this PTF is terminated.

Programmer Response: Correct the source of the error and add the PTF entry to the SMPACDS using UCLIN or re-issue the ACCFPT for this PTF.

HMA262I ERROR FORCES SYSGEN SCAN TO TERMINATE

Explanation: An error (explained in a previous message) causes the scan of the Stage I system generation file to terminate.

System Action: The system generation scan is terminated.

Programmer Response: Perform the action required depending upon the previous message and re-submit the job.

HMA263I ERROR OCCURRED IN STEP sss of JOB jjj

Explanation: sss is the step name; jjj is the job name. This message indicates the job and step in which an error occurred that was indicated by a previous message.

System Action: None.

Programmer Response: None.

HMA264I MACRO mac for PTF nnn NOT FOUND ON SMPCDS LIBRARY

Explanation: mac is the macro name; nnn is the PTF number. A macro specified in the PTF could not be located on the CDS for an ACCEPT function.

System Action: Processing for this PTF is terminated.

Programmer Response: Since this must have occurred due to a UCLIN DEL function for the macro, add the macro using the UCLIN ADD function, and process the PTF again with ACCEPT.

HMA265I ACCEPT PROCESSING COMPLETED FOR PTF nnn – NOTE ZAP ERROR.

Explanation: This message is preceded by HMA251I indicating that SUPERZAP found an error during the REP pass.

System Action: SMP processes any remaining modules, macros, or superzaps for this PTF.

Programmer Response: Check the SUPERZAP listing and manually put on the superzaps that failed.

HMA266I ERROR OCCURRED IN $\left\{\begin{array}{l}\text{LINKAGE EDITOR}\\ \text{IEBCOPY}\\ \text{ASSEMBLER}\end{array}\right\}$ **INPUT**

Explanation: This documents the type of system generation step that was being scanned for CDS update when an error (indicated by a prior message) occurred.

System Action: The HMASMP job is terminated.

Programmer Response: Correct the error and re-run the job.

HMA267I DIRECTORY SPACE EXCEEDED ATTEMPTING TO STORE xxx yyy ON lib LIBRARY

Explanation: xxx is MODULE, LOAD MODULE, MACRO, PTF, ASSEMBLY, DISTLIB, NUCLEUS, UPDTE, ZAP, MEMBER, or SYSTEM ENTRY; yyy is module, load module, macro, PTF, assembly, DISTLIB, nucleus, UPDTE, zap, member, or system entry name; lib is the library name. The number of directory blocks allocated to the data set specified by ddname was exceeded in attempting to STOW the specified member.

System Action: The member is not stored. HMASMP action is indicated by messages that follow.

Programmer Response: Enlarge the allocation for directory blocks for the indicated library and re-submit the job.

HMA268I I/O ERROR OCCURRED ATTEMPTING TO STORE xxx yyy ON lib LIBRARY

Explanation: xxx is MODULE, LOAD MODULE, MACRO, PTF, ASSEMBLY, DISTLIB, NUCLEUS, UPDTE, ZAP, MEMBER, or SYSTEM ENTRY; yyy is module, load module, macro, PTF, assembly, DISTLIB, nucleus, UPDTE, zap, member, or system entry name; lib is the library name. An input/output error has occurred on a STOW operation for the indicated member in the specified library.

System Action: The member is not stored. HMASMP action is indicated by the following messages.

Programmer Response: Correct the cause of the error by copying the file to another device and re-submit the job.

HMA269I I/O ERROR OCCURRED ATTEMPTING TO BLDL FOR xxx yyy ON lib LIBRARY

Explanation: xxx is MODULE, LOAD MODULE, MACRO, PTF, ASSEMBLY, DISTLIB, NUCLEUS, UPDTE, ZAP, MEMBER, or SYSTEM ENTRY; yyy is module, load module, macro, PTF, assembly, DISTLIB, nucleus, UPDTE, zap, member, or system entry name; lib is the library name. A BLDL operation produced an I/O error on the library specified.

System Action: The HMASMP action is indicated by the following messages.

Programmer Response: Correct the cause of the error and re-submit the job.

HMA270I PTF nnn FOUND ON SMPACDS LIBRARY – NOLIB ASSUMED

Explanation: nnn is the PTF number. During the ACCEPT function, the PTF specified was located on the SMPACDS library.

System Action: Since the PTF must already have been ACCEPTed with another system, it already exists on the DLIBs, and the NOLIB option is assumed. No updates to the DLIBs will occur.

Programmer Response: None.

HMA271I LOAD MODULE = mod

Explanation: mod is the load module name. As each MOD or ZAP type (module) entry is processed during the PTF RECEIVE processing, a list of this module's load module name(s) is made. This is done to inform you of the changes that will occur when APPLY and ACCEPT are performed.

System Action: Normal processing continues.

Programmer Response: None.

HMA272I DISTRIBUTION LIBRARY = lib

Explanation: lib is the DLIB ddname. As each MOD or ZAP type (module) entry is processed during the PTF RECEIVE processing, a list of this module's distribution library name is printed. This is done to inform you of the changes that will occur when APPLY and ACCEPT are performed.

System Action: Normal processing continues.

Programmer Response: None.

HMA273I INPUT TEXT NOT FOUND

Explanation: The input text could not be found on the PTF file or the data set specified in the TXLIB or LKLIB keyword on the module or macro header.

System Action: Processing for the PTF containing this module or macro is terminated.

Programmer Response: Correct the name of the data set in the TXLIB or LKLIB keyword or remove that keyword and insert the module text in-line. If TXLIB or LKLIB was not used, insure that the text follows the header on the PTF file.

HMA274I I/O ERROR jobname, stepname, unit address, device type, ddname, operation attempted, error description, last seek address, or block count, access method

Explanation: An input/output error occurred while accessing the named data set.

System Action: HMASMP action is described by following messages.

Programmer/Operator Response: Correct the error condition and re-submit the job.

HMA275I FOLLOWING SELECTED PTF NUMBERS NOT PROCESSED

Explanation: The following PTFs were not found on the SMPPTFIN file or the receive processing was abnormally terminated before completion of the processing of the PTFs.

System Action: None.

Programmer Response: If the PTF was not found on the SMPPTFIN file, insure that the PTF number selected on the RECEIVE statement is correct. If RECEIVE abnormally terminated, correct the error and re-run the PTF listed.

HMA276I ILLEGAL UPDATE REQUEST

Explanation: An illegal combination of UCL operations was attempted.

System Action: The member is not updated.

Programmer Response: Check the HMASMP manual for the syntax of the UCL, fix the statement in error, and re-submit the job.

HMA277I DELETE REQUEST WILL CAUSE INSUFFICIENT DATA

Explanation: Deletion of information from the CDS entry will cause the entry to have insufficient information to be restored.

System Action: The request is terminated.

Programmer Response: Probable user error. Use the UCL REP statement to replace (not delete) data from the entry. A LIST CDS (or partial list) may be helpful at this point.

HMA278I mac FLUSHED – PTF FOUND ON SMPACDS LIBRARY

Explanation: mac is the macro name. During RECEIVE, the macro is flushed because it is assumed to already exist on the distribution library.

System Action: The macro will not be replaced and the backup copy will not be saved. For an UPDTE type, IEBUPDTE will not be used.

Programmer Response: None

HMA279I CONTROL DATA SET UPDATE COMPLETE

Explanation: This message is issued in response to completion of the processing of the JCLIN statement.

System Action: None.

Programmer Response: None.

HMA280I INSUFFICIENT STORAGE FOR INCORE CDS UPDATE

Explanation: Storage was not available to perform an in-core CDS update; therefore, the JCLIN request is terminated.

System Action: The JCLIN operation is terminated.

Programmer Response: Rerun the job alloting a larger partition or region.

HMA281I DUPLICATE MODULES OCCUR WITHIN PTF

Explanation: In scanning the PTF input tape during the RECEIVE opteration, two '++MOD' statements naming the same module occurred within one PTF.

System Action: The scan of this PTF is terminated.

Programmer Response: Probable user error. Correct the input file and re-submit the job.

HMA282I CDS DIRECTORY UPDATE WILL EXCEED AVAILABLE DIRECTORY BLOCKS

Explanation: During JCLIN processing, it was determined that the number of directory blocks in the CDS will be exceeded.

System Action: The JCLIN processing is terminated.

Programmer Response: Increase the number of directory blocks for the CDS and resubmit the job.

HMA283I ERROR – ddn DD STATEMENT

MISSING { **FOR LOAD MODULE mod**
FOR MODULE mod IN PTF nnn }

Explanation: The specified DD statement (ddn) does not exist and is required.

System Action: HMASMP action is indicated by following messages.

Programmer Response: Add the required DD statement or correct the ddname.

HMA399D ENTER JULIAN DATE OR 'U' FOR HMASMP

Explanation: SMP is requesting the date to be recorded for this run.

System Action: None.

Programmer/Operator Response: Enter the date as yyddd or reply with 'U' for the system IPL date. (yy = year, ddd = day)

Data Set Requirements Chart

	RECEIVE	APPLY	ACCEPT	REJECT	RESTORE	LOG	JCLIN	UCLIN	LIST
SYSPRINT	X	X	X		X				
SMPOUT	X	X	X	X	X	X	X	X	X
SMPLOG	X	X	X	X	X	X	X	X	X
SMPCDS	X	X	X	X	X	X	X	X	X
SMPACDS	X		X						
SMPPTS	X	X	X	X	X				
SYSUT1	X	X	X		X		X		
SYSUT2	X	X	X		X				
SYSUT3	X	X	X		X				
SYSLIB	X				X				
SMPMACn	X		X	X	X				
SMPPTFIN	X								
SMPCNTL	X	X	X	X	X	X	X	X	X
SMPJCLIN							X		
tgtlib		X			X				
distlib			X		X				
lklib[1]		X	X						
txlib[1]	X	X	X						

[1] Required only if specified on the ++ MOD or ++ MAC statements.

Figure 12. Data Set Requirements Chart

Cataloged Procedure

A cataloged procedure is a set of job control statements that are placed in a partitioned data set and can be retrieved by naming it in an execute (EXEC) statement. A suggested SMP cataloged procedure is provided below. You can put it into a cataloged procedure library and use it during the execution of an SMP job.

Refer to the topic 'Data Sets' in Chapter 2 for a description of the data sets shown in this procedure.

Example of Suggested SMP Cataloged Procedure

```
//SMPJOB   PROC
//SMPSTEP  EXEC PGM=HMASMP,PARM='DATE=U'
//SMPPTS   DD   DSN=SMPPTS,DISP=OLD
//SMPOUT   DD   SYSOUT=A
//SMPLOG   DD   DSN=SMPLOG,DISP=MOD
//SMPCDS   DD   DSN=SMPCDS,DISP=OLD
//SMPACDS  DD   DSN=SMPACDS,DISP=OLD
//NUCLEUS  DD   DSN=SYS1.NUCLEUS,DISP=OLD
//SVCLIB   DD   DSN=SYS1.SVCLIB,DISP=OLD
//LINKLIB  DD   DSN=SYS1.LINKLIB,DISP=OLD
      (List additional DD statements for operating system data sets.)
//AOSB3    DD   DSN=SYS1.AOSB3,DISP=OLD
//AOS12    DD   DSN=SYS1.AOS12,DISP=OLD
//AOSD0    DD   DSN=SYS1.AOSD0,DISP=OLD
      (List additional DD statements for distribution library data sets.)
//SMPMAC1  DD   DSN=SYS1.AMODGEN,DISP=OLD
//SMPMAC2  DD   DSN=SYS1.AGENLIB,DISP=OLD
//SMPMAC3  DD   DSN=SYS1.ASAMPLIB,DISP=OLD
      (List additional SMPMACn DD statements for text data sets that may be affected by
      ++ MAC or ++ UPDTE type PTFs.)
//SYSLIB   DD   DSN=SYS1.AMODGEN,DISP=OLD
//         DD   DSN=SYS1.AMACLIB,DISP=OLD
      (List in the SYSLIB concatenation the additional DD statements that are required
      for assemblies.)
//SMPCNTL  DD   DDNAME=SYSIN
//SYSUT1   DD   UNIT=SYSDA,SPACE=(1700,(600,100))
//SYSUT2   DD   UNIT=SYSDA,SPACE=(1700,(600,100))
//SYSUT3   DD   UNIT=SYSDA,SPACE=(1700,(600,100))
//SYSPRINT DD   SYSOUT=A
```

Notes:

1. For a RECEIVE procedure, the SMPPTFIN DD statement must be added to describe the modification input. If there are no ++ MAC or ++ UPDTE modifications being processed, then the SYSLIB or SMPMACn statements are not required.
2. The distribution library DD statements are not required for a RECEIVE, APPLY, or REJECT procedure.
3. The SMPMACn and SYSLIB DD statements are not required for an APPLY procedure.
4. The SMPACDS DD statement is not required for an APPLY, REJECT, or RESTORE procedure.

Restrictions

PEMAX

PEMAX defines the maximum number of PTF elements that may exist in the CDS. A PTF element is any component (MOD, MAC, UPDTE, or ZAP) that may occur in a PTF. The default value for this is set at 50, but may be changed using the UCL.

The only need to update this will occur when more than 50 elements occur within a modification, which could occur in an ICR. The user will receive a message issued by the RECEIVE function indicating that PEMAX is exceeded. At this time he must put in the following the SMP control statements:

```
UCLIN .
REP SYS   PEMAX(nnn) .
ENDUCL .
```

nnn
is a three digit number indicating the new number of elements.

This number is used to determine the size of SMP work areas and will cause excessive storage use if set at too high a value.

Note: You should not reduce this value below the maximum number of PTF elements.

System Levels

As in the case of previous PTF application techniques, you must make certain that the correct level of the utilities that SMP uses are available. For example, incompatible linkage editor may cause a module to be placed incorrectly in a library. This could happen when using OS 20.6 to apply a PTF to 21.0. The absence of IDR support would cause flagging of the IDENTIFY statement.

IEHIOSUP

IEHIOSUP is a system dependent utility and critical to successful PTF application to OS and OS/VS1. Internally, SMP attempts to preclude use of an incorrect level of IEHIOSUP. The algorithm used to obtain IEHIOSUP is:

- If the LINKLIB DD statement is present, SMP will issue a BLDL for IEHIOSUP on that library. If the BLDL fails, IEHIOSUP will not be executed. If the BLDL is successful, that version of IEHIOSUP will be used instead of the one on the running system.

- If the LINKLIB DD statement is not present, SMP will issue a BLDL for IEHIOSUP on the running system's LINKLIB. If it is found, IEHIOSUP will be executed. If it is not found, IEHIOSUP cannot be executed.

Multiple CDS From One Set of DLIBs

SMP cannot simultaneously update the macro libraries on the DLIBs for multiple systems (each with its own CDS) that have been generated from the same set of DLIBs. The procedure to avoid this situation is to process each PTF from RECEIVE through ACCEPT for one system then for each of the others in turn.

Error Recovery

If there is an ABEND or system failure while one of the SMP application functions (RECEIVE, APPLY, REJECT, RESTORE, or ACCEPT) is executing, there are certain techniques you can use for recovery.

The messages printed on SMPOUT and SYSPRINT should be closely examined to determine the extent that the function was performed prior to the failure. If SYSOUT is lost, execute a LIST LOG. This will give you the contents of the LOG data set and that may help you determine the cause of the failure.

The descriptions below outline the procedures to follow to recover from a failure. Note that in all cases, the update control language (UCL) is always available to change PTF entries so that the proper status is maintained. Also executing a LIST LOG and LIST PTF can sometimes supply the necessary information to determine the status. If IEHLIST LISTPDS is issued for the PTS, the presence of members there can be determined.

RECEIVE

If the PTF message from RECEIVE is issued, then the PTF actually was completely stored. Execute a LIST CDS. This should reveal the presence of the PTF. This PTF is now ready for processing by the APPLY function.

If you are unsure whether or not the PTF was completely processed by RECEIVE, use the REJECT function. This will clean up the results of RECEIVE. Then PTF(s) that failed may then be processed by the RECEIVE function again.

Note that if a LIST CDS or LIST PTF is executed and you find a PTF entry with the word 'ERROR', execute the REJECT function specifying this PTF and then process it again with RECEIVE.

APPLY

If the system failed or the linkage editor returned an error code, the CDS PTF entry is marked as applied, even though application did not actually occur. By verifying the output you may determine whether the application did take place. If it is in, then there is no problem. If the PTF was marked as applied, but you are unsure of actual application, you can issue the APPLY again coding the SELECT keyword to ensure reapplication.

If the PTF is in error (indicated by an error return code from the linkage editor), you can use the RESTORE function to ensure removal of the PTF from the operating system libraries as well as the CDS.

ACCEPT

If a failure occurs during the execution of the ACCEPT function, ensure that the PTF entry in the ACDS is not marked accepted and then: (1) reissue the ACCEPT statement if the PTS members have not been deleted, or (2) reissue the RECEIVE statement and then the ACCEPT statement with the FORCE keyword if the PTS members had been deleted.

REJECT

If a failure occurs during the execution of the REJECT function, reissue the REJECT statement to ensure that the entries in the PTS and the CDS are deleted.

RESTORE

If a failure occurs during the execution of the RESTORE function, reissue the RESTORE statement to ensure proper removal of the PTF(s) specified.

How to Use the Alternate CDS (ACDS)

The alternate CDS (ACDS) contains the data about the PTFs that have been put in the DLIBs and permanent user libraries. It contains the PTF entries for the PTFs that have been processed by the ACCEPT function with the REGEN indicator set and therefore it describes the status of the DLIBs. If the SMPCDS DD statement points to the ACDS, it may be listed using the LIST PTF statement and updated using the UCL statements.

The ACDS is used to aid in the handling of multiple systems (systems generated from the same set of DLIBs) and in the handling of a complete system generation performed after PTFs have been put in the DLIBs. These two major uses of the ACDS are described below.

Multiple Systems

When more than one system is generated and PTFs are applied to both systems, care must be taken by the user to insure that regression does not occur on the DLIBs. That is, an earlier PTF against the same module or macro definition placed on the DLIB by one system may overlay (during ACCEPT processing) a later PTF placed there by another system. This interaction may be controlled by keeping all systems at relatively the same PTF level.

The SMPACDS DD statement may be included in the JCL for the RECEIVE process. SMP will use the ACDS to determine the course of macro handling during RECEIVE. If a PTF being processed by RECEIVE exists on the ACDS and *not* on the CDS, then it is assumed that the PTF must have been placed on the DLIBs by another system. Macros are then assumed to already exist on the macro library in the updated form. Therefore MAC and UPDTE types of input are flushed through by RECEIVE.

The REJECT process cannot retrieve a back-up copy of the macro because none exists. The RESTORE function should not be run for the extra systems because back-up copies of the macros do not exist and earlier versions of the modules affected will not exist.

In the case of PTFs containing macros, SMP will produce error messages that will indicate that the RESTORE function did not complete. The user must be aware that modules will be at the same level on the DLIBs as in the system because of an earlier application. The only back-off capability may be obtained through use of the LIB keyword on the RESTORE statement if a prior copy of all the required modules exists on the indicated library.

The ACCEPT function will not update distribution libraries again if the PTF already exists on the ACDS. The modules already exists in updated form there from the previous execution of ACCEPT.

After a Full System Generation

When a full system generation is performed, it is always necessary to allocate a new CDS, copy the ACDS into the new CDS using IEBCOPY, and complete the new CDS using JCLIN.

The PTF entries in the new CDS will have the REGEN indicator set indicating that the PTFs are considered to be applied to the modules in the newly generated system.

Module Conflicts During Partial System Generation

Functional difficulties can occur when applying PTFs after performing a partial (I/O device or nucleus) system generation. These difficulties occur because mutually exclusive pairs of DLIB modules can occur in the operating system depending on the options specified during the system generation and then the partial generation.

To prevent both module entries from occurring in the CDS, both module entries are deleted from the CDS using one of the SMP procedures (SMPIO or SMPNUC) before updating the CDS with the data from the partial generation with JCLIN.

The procedures are in SYS1.PROCLIB and are:

- Nucleus Generation – SMPNUC
- I/O Generation – SMPIO

These procedures may be executed using the operator START command or through JCL.

START Command

```
S     SMPNUC.id,aaa,vvv,,dsn=ddd       (Nucleus Generation)
S     SMPIO.id,aaa,vvv,,dsn=ddd        (I/O Generation)
```

id

any characters for MVT or VS2; partition number for MFT or VS1.

aaa

is the device address or device type of the CDS. The default is SYSDA.

vvv

is the volume serial number of the CDS device; this parameter is required.

ddd

is the data set name of the CDS. The default is SMPCDS.

JCL

```
//            EXEC  { SMPNUC }           (Nucleus Generation)
                    { SMPIO  }           (I/O Generation)
//IEFPROC.IEFRDER DD VOL=SER=vvv,UNIT=aaa,DSN=ddd
```

See the above description for the parameter definitions.

Note: SMPLOG DD statement points to SYSOUT=A, so it may be overridden to point to the LOG data set.

Index

Indexes to systems reference library manuals are consolidated in the *OS/VS1 Master Index,* GC24-5104, and in the *OS/VS2 Master Index,* GC28-0693. For additional information about any subject in this index, refer to other publications listed for the same subject in the *Master Index.*

Indexes to systems reference library manuals are consolidated in the *OS/VS1 Master Index,* GC24-5104, and in the *OS/VS2 Master Index,* GC28-0693. For additional information about any subject in this index, refer to other publications listed for the same subject in the *Master Index.*

OS/VS System Modification Program (SMP)

GC28-0673-1

READER'S COMMENT FORM

Your views about this publication may help improve its usefulness; this form will be sent to the author's department for appropriate action. Using this form to request system assistance or additional publications will delay response, however. *For more direct handling of such requests, please contact your IBM representative or the IBM Branch Office serving your locality.*

Possible topics for comment are:

Clarity Accuracy Completeness Organization Index Figures Examples Legibility

What is your occupation? ____
Number of latest Technical Newsletter (if any) concerning this publication: ______________
Please indicate in the space below if you wish a reply.

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. Elsewhere, an IBM office or representative will be happy to forward your comments.

Cut or Fold Along Line

GC28-0673-1

Your comments, please . . .

This manual is part of a library that serves as a reference source for system analysts, programmers, and operators of IBM systems. Your comments on the other side of this form will be carefully reviewed by the persons responsible for writing and publishing this material. All comments and suggestions become the property of IBM.

Cut or Fold Along Line

Fold Fold

First Class
Permit 81
Poughkeepsie
New York

Business Reply Mail
No postage stamp necessary if mailed in the U.S.A.

Postage will be paid by:

International Business Machines Corporation
Department D58, Building 706-2
PO Box 390
Poughkeepsie, New York 12602

Fold Fold

OS/VS System Modification Program (SMP) (S370-37) Printed in U.S.A. GC28-0673-1

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)

GC28-0673-1

OS/VS System Modification Program (SMP) (S370-37) Printed in U.S.A. GC28-0673-1

IBM®

International Business Machines Corporation
Data Processing Division
1133 Westchester Avenue, White Plains, New York 10604
(U.S.A. only)

IBM World Trade Corporation
821 United Nations Plaza, New York, New York 10017
(International)