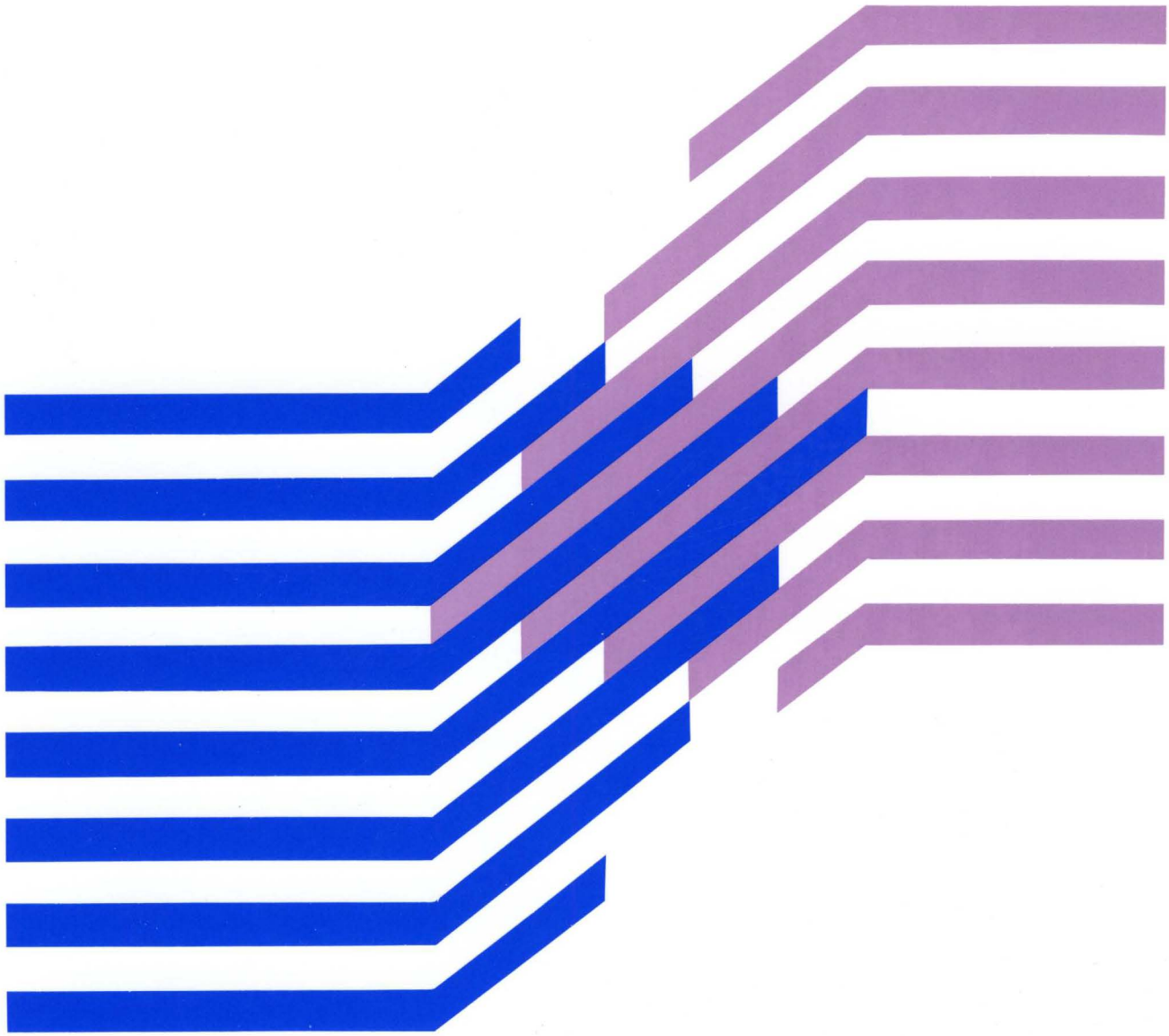
GC28-1825-3

MVS/ESA System Generation

MVS/System Product: JES2 Version 3 JES3 Version 3



GC28-1825-3

MVS/ESA
System Generation

MVS/System Product:
JES2 Version 3
JES3 Version 3

NOTE

Before using this information and the product it supports, be sure to read the general information under "Notices" on page viii.

Production of This Book

This book was prepared and formatted using the IBM BookMaster document markup language.

Fourth Edition (June, 1991)

This is a major revision of, and obsoletes, GC28-1825-02. See the Summary of Changes for the changes made to this manual. Technical changes or additions to the text and illustrations are indicated by a vertical line to the left of the change.

This edition applies to Version 3 of MVS/ESA System Product 5685-001 or 5685-002 and to all subsequent releases and modifications until otherwise indicated in new editions or Technical Newsletters. Make sure you are using the correct edition for the level of the product.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

A form for reader's comments is provided at the back of this publication. If the form has been removed, address your comments to:

IBM Corporation, Department D58
PO Box 950
Poughkeepsie, N.Y. 12602
United States of America

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Contents

Figures

Notices

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent product, program or service that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program, or service. Evaluation and verification of operation in conjunction with other products, except those expressly designated by IBM, is the user's responsibility.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Commercial Relations, IBM Corporation, Purchase, NY 10577.

Programming Interfaces

This book is intended to help customers select and define system data sets, and run the system generation process to build system libraries. It contains system data set descriptions. This book documents no programming interfaces for use by customers in writing programs that request or receive the services of MVS/ESA System Product Version 3.

Trademarks

The following terms, **DENOTED BY AN ASTERISK (*)**, used in this publication, are trademarks of the IBM Corporation in the United States and/or other countries:

- BookMaster
- IBM
- MVS/DFP
- MVS/ESA
- MVS/SP
- MVS/XA
- VTAM

About This Book

This publication provides information on how to do a system generation (sysgen) for MVS/System Product (MVS/SP) Version 3. An MVS/ESA* system consists of MVS/SP*) Version 3 and MVS/Data Facility Product (MVS/DFP*) Version 3.

Who This Book Is For

This book is intended for anyone who has system generation responsibility for their installation. Usually, this person is a systems programmer. The book assumes that the reader can:

- Code JCL statements to execute programs or cataloged procedures.
- Code in assembler language and read assembler, loader, and linkage editor output.
- Understand distribution libraries provided by IBM* and the reader's installation.

This book describes the process of creating system libraries that are customized for the data processing requirements of your installation.

How This Book Is Organized

This book is organized as follows:

Chapter 1, "Introduction": defines and discusses a sysgen. It also indicates which types of sysgen are obsolete.

Chapter 2, "Planning for an MVS System": discusses the steps to consider when planning for a sysgen.

Chapter 3, "Coding the Sysgen Statements": contains the detailed information needed to select and code the sysgen statements. Included are:

- The required and optional statements for a sysgen.
- An explanation of each sysgen statement, given alphabetically by the statement name. The format, parameters, and defaults of each statement are explained.
- An example of the coding of each statement.

Chapter 4, "Selecting and Defining the System Data Sets": describes the procedures for allocating space for the system data sets and cataloging them in the master catalog. Also included are descriptions of each of the system data sets.

Chapter 5, "Executing Sysgen": is divided into four sections:

- An explanation of Stage 1 input, processing, and output
- An explanation of Stage 2 input, processing, and output

* MVS/ESA is a trademark of the IBM Corporation
* MVS/SP is a trademark of the IBM Corporation
* MVS/DFP is a trademark of the IBM Corporation
* IBM is a trademark of the IBM Corporation

- An explanation of causes for unsatisfactory completion of sysgen and suggested restart procedures
- Examples of sysgen error, warning, and informational messages

Chapter 6, "Testing the System": discusses the procedures for testing the newly installed system using the installation verification procedure (IVP).

Chapter 7, "Using Data Sets from the Old Production System": discusses the procedure for substituting data sets from the old production system for those in the newly installed system.

Chapter 8, "Installing ISMF": discusses how to install the interactive storage management facility (ISMF).

Appendix A, "Installing the DFP Starter Set": explains how to install the Data Facility Product (DFP) Starter Set.

Appendix B, "Sysgen Example": presents an example that illustrates Stage 1 input statements for a typical JES2 sysgen.

Appendix C, "Making ISAM Available on a System with DFP 3.1.0 Installed": explains how to make the Index Sequential Access Method (ISAM) available on a system with DFP 3.1.0 installed.

Related Information

Where necessary, this book references information in other books, using shortened versions of the book title. The following tables show the shortened titles, complete titles, and order numbers of the books you might need while you are using this book.

The access method services, catalog, and VSAM books you might need are:

Short Title Used in This Book	Title	Order Number
Access Method Services Reference	*MVS/ESA Integrated Catalog Administration: Access Method Services Reference*	SC26-4500
Access Method Services Reference	*MVS/ESA VSAM Catalog Administration: Access Method Services Reference*	SC26-4501
Catalog Administration Guide	*MVS/ESA Catalog Administration Guide*	SC26-4502
VSAM Administration Guide	*MVS/ESA VSAM Administration Guide*	SC26-4518
Macro Instruction Reference	*MVS/ESA VSAM Administration: Macro Instruction Reference*	SC26-4517
DFP: Managing Catalogs	*MVS/DFP Version 3 Release 2: Managing Catalogs*	SC26-4555

The Assembler H and JCL books you might need are:

Short Title Used in This Book	Title	Order Number
Programming: Guide	*Assembler H Version 2 Application Programming: Guide*	SC26-4036
Language Reference	*Assembler H Version 2 Application Programming: Language Reference*	GC26-4037
JCL User's Guide	*MVS/ESA JCL User's Guide*	GC28-1830
JCL Reference	*MVS/ESA JCL Reference*	GC28-1829

The customization book you might need is:

Short Title Used in This Book	Title	Order Number
DFP: Customization	*MVS/ESA Data Facility Product Version 3: Customization*	SC26-4504

The data management and utilities books you might need are:

Short Title Used in This Book	Title	Order Number
Utilities	*MVS/ESA Data Administration: Utilities*	SC26-4516
System—Data Administration	*MVS/ESA System—Data Administration*	SC26-4515
Data Administration Guide	*MVS/ESA Data Administration Guide*	SC26-4505
Storage Administration Reference	*MVS/ESA Storage Administration Reference*	SC26-4514

The diagnosis books you might need are:

Short Title Used in This Book	Title	Order Number
DFP: Diagnosis Guide	*MVS/ESA Data Facility Product Version 3: Diagnosis Guide*	LY27-9550
	MVS/XA Data Facility Product* Version 2: Diagnosis Guide	LY27-9521
DFP: Diagnosis Reference	*MVS/ESA Data Facility Product Version 3: Diagnosis Reference*	LY27-9551
	MVS/XA Data Facility Product Version 2: Diagnosis Reference	LY27-9530

* MVS/XA is a Trademark of the IBM Corporation

The Interactive Storage Management Facility (ISMF) book you might need is:

Short Title Used in This Book	Title	Order Number
ISMF User's Guide	*MVS/ESA Interactive Storage Management Facility User's Guide*	SC26-4508

The messages and codes books you might need are:

Short Title Used in This Book	Title	Order Number
System Codes	*MVS/ESA Message Library: System Codes*	GC28-1815
System Messages	*MVS/ESA Message Library: System Messages,* Volumes 1 and 2	GC28-1812 and 1813

The MVSCP and IOCP books you might need are:

Short Title Used in This Book	Title	Order Number
MVSCP Guide	*MVS/ESA MVS Configuration Program Guide and Reference*	GC28-1817
IOCP Guide	*IBM ES/3090 Processor Complex Input/Output Configuration Program User's Guide and Reference*	SC38-0066

The network management book you might need is:

Short Title Used in This Book	Title	Order Number
NPP General Information	*Network Programming Product General Information*	GC30-3350

The planning book you might need is:

Short Title Used in This Book	Title	Order Number
DFP: Planning Guide	*MVS/ESA Data Facility Product Version 3: Planning Guide*	SC26-4513

The system programming, initialization, tuning, and debugging books you might need are:

Short Title Used in This Book	Title	Order Number
SMP/E User's Guide	*System Modification Program Extended User's Guide*	SC28-1302
MVS/ESA Program Directory: JES3	*MVS/ESA Program Directory: JES3*	GC28-1017
System Commands	*MVS/ESA Operations: System Commands*	GC28-1826
SPL Application Development Guide	*MVS/ESA System Programming Library: Application Development Guide*	GC28-1852
31-Bit Addressing	*MVS/ESA System Programming Library: Application Development - 31-Bit Addressing*	GC28-1820
Initialization and Tuning	*MVS/ESA System Programming Library: Initialization and Tuning*	GC28-1828
Service Aids	*MVS/ESA Service Aids*	GC28-1844
SPL Application Development Macro Reference	*MVS/ESA System Programming Library: Application Development Macro Reference*	GC28-1857
SMF	*MVS/ESA System Programming Library: System Management Facilities*	GC28-1819
IPCS User's Guide	*MVS/ESA Interactive Problem Control System (IPCS) Command Reference*	GC28-1833
SYS1.LOGREC Error Recording	*MVS/ESA SYS1.LOGREC Error Recording*	GC28-1845
IPCS Command Reference	*MVS/ESA Interactive Problem Control System (IPCS) Command Reference*	GC28-1834
System Modifications	*MVS/ESA System Programming Library: System Modifications*	GC28-1831
Planning: Dump and Trace Services	*MVS/ESA Planning: Dump and Trace Services*	GC28-1838

The TSO books you might need are:

Short Title Used in This Book	Title	Order Number
TSO/E Command Reference	*TSO Extensions Command Reference*	SC28-1307
TSO/E Program Directory	*TSO Extensions Program Directory*	LC28-1284
TSO/E Customization	*TSO Extensions Customization*	SC28-1380
TSO/E Administration	*TSO Extensions Administration*	SC28-1356

Summary of Changes

Summary of Changes
for GC28-1825-3
MVS/System Product Version 3 Release 1.3

This book contains the following new and changed information.

New Information

"Master Catalog" on page 4-16 shows that, for faster response, the VSAM master catalog can point to data set entries in user catalogs.

Optimal space utilization block sizes for the 3390 direct access storage device (DASD) have been added to appropriate data set descriptions in Chapter 4, "Selecting and Defining the System Data Sets."

Changed Information

Miscellaneous technical and editorial changes have been made.

Summary of Changes
for GC28-1825-2
MVS/System Product Version 3 Release 1.3

This book contains the following new and changed information.

New Information

Chapter 4, "Selecting and Defining the System Data Sets" describes SYS1.MODGEN, a new data set.

Changed Information

Miscellaneous technical and editorial changes have been made.

Summary of Changes
for GC28-1825-1
MVS/System Product Version 3 Release 1.0e

This book contains the following new and changed information.

New Information

Chapter 4, "Selecting and Defining the System Data Sets" describes SYS1.ISAMLPA, a new data set.

Appendix A, "Installing the DFP Starter Set" explains how to install the Data Facility Product (DFP) Data Set.

Appendix C, "Making ISAM Available on a System with DFP 3.1.0 Installed" explains how to make the Index Sequential Access Method (ISAM) available on a system with DFP 3.1.0 installed.

Changed Information

Miscellaneous technical and editorial changes have been made.

Summary of Changes
for GC28-1825-0
MVS/System Product Version 3 Release 1.0

This book contains information previously presented in *MVS/XA Installation: System Generation* , GC26-4148. The following summarizes the changes to that information.

New Information

Chapter 4, "Selecting and Defining the System Data Sets" now describes the new system data set SYS1.MIGLIB.

Chapter 7, "Using Data Sets from the Old Production System" now includes the SYSCATLG member of SYS1.NUCLEUS, which will be available with MVS/DFP Version 3 for DFP 3.1.

Chapter 8, "Installing ISMF" reflects product releases that support new interactive storage management facility function.

Changed Information

Miscellaneous technical and editorial changes have been made throughout.

Deleted Information

Obsolete sysgen statements have been removed from Chapter 3, "Coding the Sysgen Statements".

Chapter 1. Introduction

System generation (usually referred to as sysgen) is a process that builds system libraries that reflect your operating system requirements. The sysgen process selects modules from the IBM distribution libraries (DLIBs), optionally combines them with installation-provided routines from one or more installation-specified partitioned data sets, and places the modules and routines in the appropriate system libraries.

Sysgen

Sysgen is the process of creating system libraries customized to the data processing requirements of your installation. To do a sysgen,

- Define (allocate space for, and catalog) the system data sets. Some of the system data sets are required in all installations; the need for others depends on the specific products and options you select.
- Execute the sysgen process as a series of jobs.

An I/O device generation—usually called IOGEN—is no longer part of sysgen. The function of an IOGEN is now provided by the MVS configuration program, various PARMLIB members, and SAMPLIB members.

An eligible device table (EDT) generation—usually called EDTGEN—is no longer part of sysgen. The function of an EDTGEN is now provided by the MVS configuration program.

Execution of Sysgen

The sysgen process runs as a series of jobs in two stages:

- **Stage 1** uses the information you code on the sysgen statements to create a job stream consisting of JCL statements and control statements, as well as a documentation listing. This job stream is the input to Stage 2, and to the SMP JCLIN function.
- **Stage 2** uses the job stream created in Stage 1 to link-edit and copy modules from the DLIBs into the new system data sets. The output from Stage 2 is the new or updated MVS/ESA system libraries and a documentation listing.

Execute the jobs that comprise Stage 1 and Stage 2 on an existing system (called the generating system). To generate or update a system that includes MVS/SP Version 3, the generating system must be an MVS/370 system at a Release 3.8 level, with or without MVS/System Product Version 1 or MVS/System Extensions installed, or an MVS/XA system.

Figure 1-1 on page 1-2 illustrates the sysgen process. This process is only part of the MVS installation process. Sysgen works with the following programs to create or update an MVS system:

- System Modification Program Extended (SMP/E)
- MVS configuration program
- Input/Output Configuration Program (IOCP)

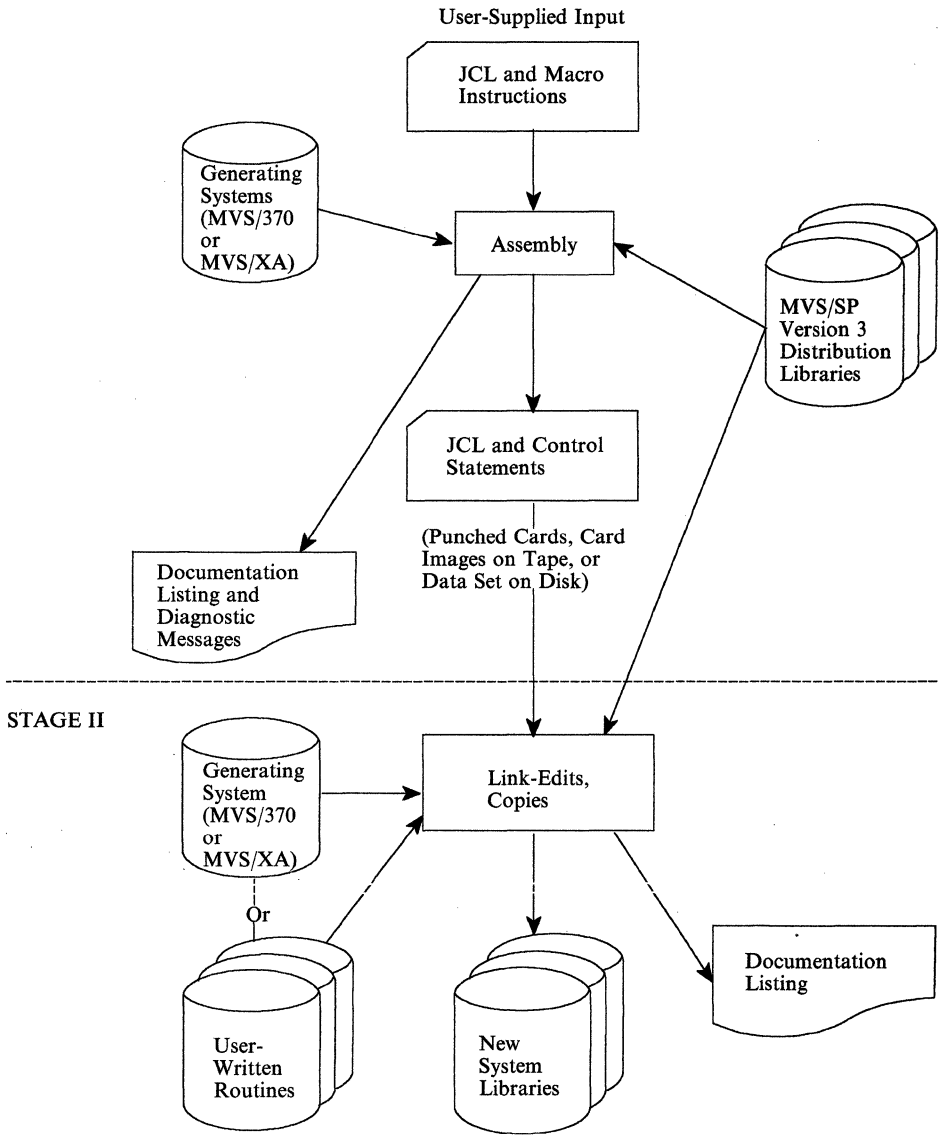


Figure 1-1. The Sysgen Process

Additionally, to build a complete MVS system, use an editor or utility program to build the required PARMLIB members.

The next three topics describe the purpose of SMP/E, the MVS configuration program, and IOCP.

System Modification Program (SMP/E or SMP)

To create or update an MVS system, you need the System Modification Program (SMP/E).

Note: In this book, the acronyms SMP and SMP/E both refer to System Modification Program Extended (SMP/E).

Before doing a sysgen, use the ACCEPT function of SMP/E to incorporate into your existing DLIBs the products you are installing. Also use SMP/E to incorporate prerequisite and corequisite service into the DLIBs.

When you use SMP/E to update your DLIBs, SMP/E ensures that corequisites or prerequisites are present on the DLIBs for the products you are installing. To do this, SMP/E maintains two data sets:

- The control data set (target zone), which reflects the contents of the system libraries
- The alternate control data set (DLIB zone), which reflects the contents of the DLIBs

After a sysgen, use the SMP JCLIN function to update the control data set to reflect changes in the system libraries.

For further information about SMP, see *SMP/E User's Guide*.

MVS Configuration Program

The MVS configuration program provides much of the function that was provided by an IOGEN as well as all of the function that was provided by an EDTGEN. The primary sources of input to the MVS configuration program are IODEVICE and UNITNAME statements. The output consists of a set of load modules that MVS uses to manage the I/O configuration.

You execute the MVS configuration program to define the I/O configuration to MVS. Required in the definition are the:

- I/O devices attached to control units
- Esoteric specifications for the I/O devices
- Device preference table specification
- NIP console devices

For additional information about the MVS configuration program, see *MVSCP Guide*.

Input/Output Configuration Program (IOCP)

Channel operations are controlled by the I/O subsystem in the processor complex running in extended architecture mode. To control the channel operations, the I/O subsystem requires specific data about the hardware I/O configuration.

You execute IOCP to define the I/O configuration for use by the I/O subsystem. Required in this definition are the:

- Channel paths on the processor complex
- Control units attached to the channel paths
- I/O devices attached to the control units

Both MVS and the hardware require data about the I/O configuration. To provide this data, use a common input stream for both IOCP and the MVS configuration program; that is, include IOCP statements in the MVS configuration program input stream to ensure identical definitions of software and hardware I/O configurations. For testing purposes, execute the MVS configuration program before IOCP. The MVS Configuration is described in *MVSCP Guide*.

Chapter 2. Planning for an MVS System

Listed below are the steps to consider when you plan to build and install an MVS system. Note that you can do several of the steps in sequences other than those shown. Technically, some of the steps are unnecessary, or you can do them in different ways.

1. Ensure that you provide the required environment for running sysgen.
2. Provide for backup and recovery.
3. Incorporate the products being installed into the DLIBs.
4. Initialize the DASD volumes that will contain the system data sets.
5. Define (allocate space for, and catalog) the system data sets.
6. Code the sysgen statements.
7. Prepare to add user-written routines during sysgen.
8. Execute sysgen.
9. Create a new SMP/E control data set to reflect the new system libraries.
10. Install code not supported by the sysgen process.
11. Code the MVS configuration program and IOCP statements.
12. Execute the MVS configuration program.
13. Execute IOCP.
14. Build required PARMLIB members.
15. Make backup copies of the new system data sets.
16. Make ISMF execution libraries available to TSO/E users.
17. After initial program load (IPL), test the system.

The following topics describe each step.

Providing the Required Environment for Sysgen

You can generate a system that includes MVS/SP Version 3 on an MVS/370 system or on an MVS/XA system. The MVS/370 system must be based on OS/VS2 MVS Release 3.8, with or without MVS/System Extensions or MVS/System Product Version 1 installed. The generating system must also support the devices on which the system libraries and DLIBs will reside; that is, if you are planning to use the IBM 3375, 3380, or 3390 direct access storage device (DASD) for the system libraries, ensure that the appropriate device support is in the generating system.

If you define the master catalog as an integrated catalog facility catalog, the generating system must contain integrated catalog facility support through previous installations of either Data Facility Extended Function (DFEF) or Data Facility Product (DFP). If you do not already have integrated catalog facility support, define the new master catalog as a VSAM catalog and convert it to integrated after testing the newly built system.

Specific levels of certain programs (such as SMP/E and Assembler H) are required for a sysgen. The program directories shipped with MVS/SP Version 3 and MVS/XA DFP Version 2.3 or MVS/DFP Version 3 identify programs required for sysgen and the required levels of those programs.

In addition, have the following programs available on the generating system:

- The AMBLIST utility in DFP, to help analyze incomplete or incorrect output from selected steps of the sysgen process
- Data Facility Data Set Services (DFDSS), to make a backup copy of the MVS/ESA system libraries created by sysgen.

Providing a Backup Copy of the Existing System

Before you update the DLIBs, you must copy them. The new system replaces and, therefore, completely deletes from the existing DLIBs the base control program (BCP) portion of your environment's system. Similarly, DFP deletes from the existing DLIBs the routines it replaces. Once you incorporate DFP and MVS/SP Version 3 into your DLIBs, there is no simple way to restore the DLIBs to their original contents. You cannot use the SMP RESTORE function to restore DLIBs to their previous level.

Updating the DLIBs

In this book, the acronyms SMP and SMP/E both refer to System Modification Program Extended (SMP/E).

Sysgen creates new or updated system libraries from the DLIBs. Before doing a sysgen, use the SMP ACCEPT function to incorporate into the existing DLIBs the products you are installing. To avoid EQUAL questions, all services must be accepted or restored before a complete sysgen is done. See *SMP/E User's Guide* or for details on using the SMP ACCEPT function.

The DLIBs contain modules and macros that Stage 2 copies and link-edits into the appropriate system data sets. These can contain:

- The operating system program options
- Utility programs
- Data management routines
- Error recovery routines
- Job management routines
- Task management routines
- Problem determination routines
- Teleprocessing routines
- ISMF routines
- ISMF panels, messages, and tables
- System parameter lists
- Cataloged procedures
- Macro definitions for the system macros that the assembler uses.

The DLIBs also contain macros that are used during the Stage 1 assembly.

Initializing the DASD Volumes

Before installing the new MVS system, initialize the DASD volumes that are to contain the new and updated system data sets. DASD volume initialization is the process of writing home addresses, a volume label, and a volume table of contents (VTOC) on a DASD volume. In addition, write the IPL text on the DASD volume that will become the system residence volume for the new system.

Initializing DASD volumes involves the following steps:

1. Vary the DASD volumes offline
2. Execute the appropriate Device Support Facilities control statements
3. Mount the DASD volumes.

Defining the System Data Sets

You can define the system data sets during sysgen using the DATASET statement, or before sysgen using JCL and/or access method services.

For more information on the data sets, see Chapter 4, "Selecting and Defining the System Data Sets" on page 4-1. Details on coding the DATASET statement are in Chapter 3, "Coding the Sysgen Statements" on page 3-1.

Coding the Sysgen Statements

Chapter 3 describes the sysgen statements.

Adding User-Written Routines during Sysgen

During sysgen, you can include your own macros, parameters, procedures, appendages, and routines in the system data sets using the DATASET statement described in Chapter 3.

Executing Sysgen

Chapter 5, "Executing Sysgen" on page 5-1, describes the input, processing, and output of Stage 1 and Stage 2 of the sysgen process. Chapter 5 also outlines the specific jobs necessary for a sysgen.

Creating a New Control Data Set

After doing a sysgen, create a new control data set to reflect the contents of the new system data sets. To do this, you:

- Allocate a new target zone (SMP/E)
- Copy the DLIB zone or ACDS (created by the SMP ACCEPT function when you updated the DLIBs) into the new control data set
- Execute the SMP JCLIN function using the Stage 1 output to update the control data set to reflect the new system libraries.

Installing Code Not Supported by Sysgen

Some products require an installation procedure separate from sysgen. For each product you install, consult the program directory for any specific installation steps.

Coding the MVS Configuration Program and IOCP Statements

To ensure that the I/O configuration defined to MVS by the MVS configuration program matches the I/O configuration defined to the hardware by IOCP, combine the IOCP statements with the MVS configuration program statements to form a combined input stream for use by both the MVS configuration program and IOCP.

For more information about coding the MVS configuration program statements, see *MVSCP Guide*. For more information about coding IOCP statements, see *IOCP Guide*.

Executing the MVS Configuration Program

Always use the latest level of the MVS configuration program, which is contained in SYS1.LINKLIB of the system that you are generating.

Execute the MVS configuration program prior to executing IOCP. Both perform validity checks on the input; however, more of the input is related to the MVS configuration program than to IOCP.

For details on executing the MVS configuration program, see *MVSCP Guide*.

Executing IOCP

Always use the latest level of IOCP.

For details on executing IOCP, see *IOCP Guide*.

Building PARMLIB Members

Build a CONSOLxx PARMLIB member to define your console configuration. If you install user-written SVC routines, you may need to update an IEASVCxx PARMLIB member.

In addition, edit the following PARMLIB members:

- CLOCKxx
- IEASYSxx
- IEFSSNxx

Also, you may want to build the following members:

- PFKTABxx
- SCHEDxx
- CSVLLxx
- COFVMFxx
- IEAAPFxx
- IGDSMSxx
- LPALSTxx

For additional information on PARMLIB members, see *Initialization and Tuning*.

Making a Backup Copy of the New System Data Sets

For backup and recovery, make a copy of the new system data sets created during sysgen.

Making ISMF Available to TSO/E Users

After you have generated the system, you can use the steps described in Chapter 8, "Installing ISMF" on page 8-1 to make ISMF available to TSO/E users.

Making ISAM Available

After you have generated the system, you can use the steps described in Appendix C, "Making ISAM Available on a System with DFP 3.1.0 Installed" on page C-1 to allow the use of ISAM.

Testing the System

Use the installation verification procedure (IVP) to test the new system. For information on using IVP, see Chapter 6, "Testing the System" on page 6-1.

Chapter 3. Coding the Sysgen Statements

This chapter contains the detailed information needed to select and code the sysgen statements. Included in this chapter are:

- An explanation of the conventions used in this book to show the statements.
- A list of the sysgen statements.
- An alphabetic listing of the statements. In this section, the function, format, parameters, and defaults of each statement are described, and a coding example of each is given.

Rules for Coding Sysgen Statements

The rules for coding sysgen statements are those of the assembler language. The following paragraphs are a summary of those rules, as stated in *Language Reference* and *Programming: Guide*.

Sysgen statements have the following standard format:

Name	Operation	Operand	Comments
Name	Statement name	Required and optional parameters	Optional

Name symbolically identifies the statement. If included, it contains from 1 through 8 alphameric characters, the first of which must be alphabetic. Begin the name in the first position of the statement and follow it with one or more blanks. Sysgen ignores the name field.

Operation identifies the statement. Precede it and follow it with one or more blanks.

Operand contains parameters separated by commas. You end this field by placing one or more blanks after the last parameter. In most sysgen statements, you use keyword parameters in the operand field. A parameter consists of a keyword followed by an equal sign (=) and the keyword value. The keyword value may be a single value or a list of values. If it is a list, separate the values with commas and enclose the list in parentheses.

Comments can be included on a sysgen statement. If included, separate them from the last parameter of the operand field with one or more blanks. Use an entire statement for a comment by placing an asterisk in the first column of each statement. A statement that has no parameters cannot have comments.

The rules for coding sysgen statements are:

- Code sysgen statements in columns 1 through 71.
- Continue a statement that exceeds 71 columns onto one or more additional statements by placing a nonblank character in column 72 to indicate continuation.

- Interrupt a sysgen statement either at column 71 or after any comma that separates parameters. Begin the continued portion in column 16 of the following statement.
- Comments may appear on every line of a continued statement.
- Use columns 73 through 80 to code any identification and/or sequence characters.

Describing Sysgen Statements

The notation used to define the syntax of the sysgen statements is as follows:

1. The set of symbols listed below is used to define the syntax, but never use them on a statement.
   - logical OR |
   - braces { }
   - brackets [ ]
   - underscore ___
   - ellipsis ...
2. Use uppercase letters, numbers, and the set of symbols shown in the following list on a statement exactly as shown in the syntax.
   - comma ,
   - equal sign =
   - parentheses ( )
3. Lowercase letters appearing in the syntax represent variables for which you substitute specific information on a statement.
4. Items separated by the logical OR represent alternative items. Select only **one** of the items.
5. Braces group related, alternative items. Select only **one** of the items.
6. Brackets also group related, alternative items; however, everything within the brackets is optional. If you do not **explicitly** specify one of the items, the result is a **null** specification.
7. An underscore indicates a default alternative. If you select an underscored alternative, you need not explicitly specify it on a statement. The absence of an explicit specification is an **implicit** specification of the default alternative.
8. An ellipsis indicates that you may specify the preceding item or group of items:
   - zero times,
   - once, or
   - more than once in succession.

List of Statements

The sysgen statements are as follows:

Statement	Description
DATAMGT	A required statement that you use to specify the inclusion of optional access methods, graphic programming services, Mass Storage System (MSS), and TSO commands.
DATASET	A required statement that you use to define system data sets and include user-written routines in the system data sets.
GENERATE	A required statement that you use to specify the system residence volume and JCL information for Stage 2 jobs.
JES	Beginning with the JES3 component of MVS/System Product Version 3, Release 1.3 (FMID HJS3313), the JES statement no longer installs JES3. To install JES3, use SMP/E commands, either the GENERATE command or the RECEIVE, APPLY, and ACCEPT commands.

DATAMGT

Required

Use the DATAMGT statement to specify the inclusion of optional access methods, graphic programming services, Mass Storage System (MSS), and TSO commands into the system.

Sysgen always includes the following access methods in the system:

BDAM	basic direct access method
BPAM	basic partitioned access method
BSAM	basic sequential access method
QSAM	queued sequential access method
VSAM	virtual storage access method

For information on the optional access methods, see:

NPP General Information

Data Administration Guide

[symbol]	DATAMGT	ACSMETH=({TCAM[,method]...| VTAM[,method]...| TCAM,VTAM[,method]...}) [GRAPHCS=([GSP][,PORRTNS])] [MSS={NO|YES}] [TSOCMDS={NO|YES}] IND TABLE UCSDFLT

ACSMETH = method
specifies the inclusion of one or more of the following access methods:

BTAM basic telecommunication access method (requires that you install BTAM/System Product)

GAM graphics access method (including Graphic Access Method/System Product)

ISAM basic and queued indexed sequential access methods

TCAM telecommunications access method (requires that you install ACF/TCAM Version 2)

Note: The TCAM option of the DATAMGT macro is no longer supported with TCAM 3.1.0 or higher.

VTAM* virtual telecommunications access method (requires that you install ACF/VTAM Version 2)

If you specify VTAM, define SYS1.VTAMLIB and include it in the IEAAPFxx member of PARMLIB.

GRAPHCS =
specifies the inclusion of graphic programming services.

GSP
specifies the inclusion of the graphic subroutine package in SYS1.LINKLIB.

PORRTNS
specifies the inclusion of the problem oriented routines (PORS) in SYS1.LINKLIB.

Note: If you install GAM/SP, the PORS will not be used. If these routines are already on the system it will *not* be necessary to remove them.

MSS = {*NO*|YES}
specifies whether or not to include the Mass Storage System (MSS).

* VTAM is a trademark of the IBM Corporation

TSOCMDS = {**NO**|*YES*}

specifies whether or not to include the full TSO command system. (See *TSO/E Command Reference*.) If you explicitly or implicitly specify YES, define SYS1.CMDLIB. If you specify NO, the TSO command system is included, but with limited TSO command processing functions.

If you explicitly or implicitly specify TSOCMDS=YES, specify either VTAM or TCAM on the ACSMETH= parameter; or you will receive the following Mnote during Stage 1:

```
5,***IEIDAT106 TCAM or VTAM MUST BE SPECIFIED FOR
     TSOCMDS=YES
```

IND

The IND parameter is no longer supported during sysgen, but is allowed in the Stage 1 deck for compatibility. Maintaining the IND parameter on the DATAMGT statement in your Stage 1 deck causes the following Mnote during Stage 1:

```
0,IND PARAMETER IGNORED.  SPECIFY VTAM ON
  ACSMETH PARAMETER TO INSTALL VTAM
```

TABLE

The TABLE parameter is no longer supported during sysgen, but is allowed in the Stage 1 deck for compatibility. Maintaining the TABLE parameter on the DATAMGT statement in your Stage 1 deck causes the following Mnote during Stage 1:

```
0,TABLE PARAMETER IGNORED - ALL IMAGE SETS
  INCLUDED
```

UCSDFLT

The UCSDFLT parameter is no longer supported during sysgen, but is allowed in the Stage 1 deck for compatibility. Maintaining the UCSDFLT parameter on the DATAMGT statement in your Stage 1 deck causes the following Mnote during Stage 1:

```
0,UCSDFLT PARAMETER IGNORED - ALL UCS IMAGES ARE
  DEFAULTS
```

Example: This example specifies the following options:

- The inclusion of BTAM, TCAM, and ISAM.
- The inclusion of the graphic subroutine package and problem oriented routines in SYS1.LINKLIB.
- The inclusion of MSS.
- The inclusion of the full TSO command system (the default value).

```
DATAMGT   DATAMGT   ACSMETH=(BTAM,TCAM,ISAM),              X
                    GRAPHCS=(PORRTNS,GSP),MSS=YES
```

DATASET

Required

Use the DATASET statement to define (allocate space for, and catalog) system data sets (including the master catalog) during sysgen. Additionally, you can use the DATASET statement to include user-written routines, macros, parameters, procedures, and authorized EXCP appendages in the system data sets.

Note that you must catalog some of the optional system data sets, if certain conditions exist, even if you do not allocate space for them. You can use the DATASET statement to affect the cataloging.

Prior to sysgen, you can predefine the system data sets. Use access method services to catalog, and allocate space for, VSAM data sets. Use JCL to catalog and access method services to allocate space for non-VSAM data sets.

If you predefine a system data set and do not allocate space for it, and on the DATASET statement specify SPACE=(CYL,(0)), Stage 2 neither catalogs the data set nor allocates space for it.

Page and Swap Data Sets

When you specify a page or swap data set on a DATASET statement, Stage 2 allocates space for it, catalogs it, and includes its name in IEASYS00 in SYS1.PARMLIB.

User-Written Routines

Ensure that user-written routines are members of cataloged partitioned data sets. Except for some routines that will reside in SYS1.NUCLEUS, executable code must be in load module form; that is, you must have assembled and link-edited each routine.

If a routine can reside above 16 megabytes or if it supports 31-bit addressing, specify AMODE and RMODE using the linkage editor or in the assembler language source code. For a description of the control statements required by the assembler and linkage editor, see *Language Reference* and *31-Bit Addressing*.

EXCP Appendages

You can include EXCP appendages written for use by unauthorized programs by specifying the name or names of the appendages in the ABEAPP, CHEAPP, EOEAPP, PCIAPP, or SIOAPP parameters when defining the system data set into which you want them included. Stage 2 includes the EXCP appendages in the specified system data sets and includes their names and types in IEAAPP00 in SYS1.PARMLIB. Even if you want to include the appendages after sysgen, specify their names on a DATASET statement. Stage 2 includes just the name and type of the appendage in IEAAPP00.

Although you can specify as many as 84 appendages for any **one** of the parameters, and the values you assign to each of these 5 parameters must be unique from the other four, the naming convention limits the **total** number of appendages to 144 for the sum of the appendages you specify in the 5 parameters.

Information about EXCP appendages can be found in *DFP: Customization*.

[symbol]	DATASET	{system data set identifier| PAGEDSN=name| SWAPDSN=name| SMFDSN=(suffix[,suffix]...)} [ABEAPP=(suffix[,suffix]...)][1] [CHEAPP=(suffix[,suffix]...)][1] [EOEAPP=(suffix[,suffix]...)][1] [MEMBERS=(name[,name]...)][1] [NAME={master catalog name|duplex data set name}] [PCIAPP=(suffix[,suffix]...)][1] [PDS=name] [RESIDNT=(name[,name]...)][1] [SIOAPP=(suffix[,suffix]...)][1] [SPACE=({blocksize|CYL|TRK}, (quantity))] [VOL=({volser|SYSRES},{genname| 3380})]

ABEAPP = suffix
specifies a user-written abnormal-end appendage. Specify an alphameric value, from WA through Z9, that is the last 2 characters of the appendage name; Stage 2 automatically assigns the first 6 characters (IGG019) of the name. You may specify as many as 84 appendages. Specify this parameter only for SYS1.LPALIB or SYS1.SVCLIB. If you specify this parameter, also specify the PDS parameter.

CHEAPP = suffix
specifies a user-written channel-end appendage. Specify an alphameric value, from WA through Z9, that is the last 2 characters of the appendage name; Stage 2 automatically assigns the first 6 characters (IGG019) of the name. You may specify as many as 84 appendages. Specify this parameter only for SYS1.LPALIB or SYS1.SVCLIB. If you specify this parameter, also specify the PDS parameter.

EOEAPP = suffix
specifies a user-written end-of-extent appendage. Specify an alphameric value, from WA through Z9, that is the last 2 characters of the appendage name; Stage 2 automatically assigns the first 6 characters (IGG019) of the name. You may specify as many as 84 appendages. Specify this parameter only for SYS1.LPALIB or SYS1.SVCLIB. If you specify this parameter, also specify the PDS parameter.

[1] The validity of these parameters depends on the type of system data set you define.

MEMBERS = name
specifies the member or members of a cataloged partitioned data set you are defining. Each name cannot exceed 8 alphameric characters. The first character must be alphabetic. You may specify as many as 20 members; however, the sum of the names specified by MEMBERS and RESIDNT cannot exceed 20.

Except for SYS1.NUCLEUS, if you specify this parameter, also specify the PDS parameter. Stage 2 selectively copies each member into the system data set. For SYS1.NUCLEUS, if you specify the PDS parameter, but neither MEMBERS nor RESIDNT, Stage 2 **copies** the members of the partitioned data set into SYS1.NUCLEUS as individual members. If you specify both the PDS and MEMBERS parameters, Stage 2 **link-edits** the members into the nucleus being generated (IEANUC01).

Do not specify the same **name** in both the MEMBERS parameter and the RESIDNT parameter.

Figure 3-1 on page 3-15 identifies the system data sets for which you may specify this parameter, and the form that the members must be in for inclusion in the system data sets.

NAME = name
specifies the unique name of the master catalog or the unique name of the duplex page data set.

For duplex page data sets, the name cannot exceed 44 characters. The characters can be alphameric, @, #, $, or special characters, such as the hyphen. Separate names containing more than 8 characters by means of periods between segments of from 1 to 8 characters; periods count toward the maximum of 44 characters. The first character of any name or name segment must be an alphabetic or national character.

For the master catalog, specify the name as ICFCATLG (for an integrated catalog facility master catalog) or VSCATLG (for a VSAM master catalog).

Specify this parameter for the master catalog, even if you predefine it.

PAGEDSN = name
specifies the name of a page data set. The name cannot exceed 44 characters. The characters can be alphameric, @, #, $, or special characters, such as the hyphen. Separate names containing more than 8 characters by means of periods between segments of from 1 to 8 characters; periods count toward the maximum of 44 characters. The first character of any name or name segment must be an alphabetic or a national character.

Note: Stage 2 does not append the index value you specify on the GENERATE statement to the page data set name you specify in this parameter.

Define at least 3 page data sets before or during sysgen, but not more than 25 using both this parameter and access method services. (You can define additional page data sets after sysgen.)

For additional information, see Figure 3-1 on page 3-15.

PCIAPP = suffix
specifies a user-written program-controlled interrupt (PCI) appendage. Specify an alphameric value, from WA through Z9, that is the last 2 characters of the appendage name; Stage 2 automatically assigns the first 6 characters (IGG019) of the name. You may specify as many as 84 appendages.

Specify this parameter only for SYS1.LPALIB or SYS1.SVCLIB. However, if your program runs in a V=R address space and uses a PCI appendage, do not include the PCI appendage, and any appendage it refers to, in SYS1.LPALIB. Instead, include those appendages in either SYS1.SVCLIB or the fixed link pack area.

If you specify this parameter, also specify the PDS parameter.

PDS = name
specifies the complete, fully qualified name of a cataloged partitioned data set that contains user-written routines you want included in a system data set. The name cannot exceed 44 characters. The characters can be alphameric, @, #, $, or special characters, such as the hyphen. Separate names containing more than 8 characters by means of periods between segments of from 1 to 8 characters. SYS1. and periods count toward the maximum of 44 characters. The first character of any name or name segment must be an alphabetic or a national character. Catalog the partitioned data set you specify in the generating system.

Except for SYS1.NUCLEUS, if you specify this parameter, also specify the MEMBERS or RESIDNT parameters, or both. Stage 2 selectively copies each member of the partitioned data set into the system data set. For SYS1.NUCLEUS, if you specify this parameter and not MEMBERS, Stage 2 **copies** the members of the partitioned data set into SYS1.NUCLEUS as individual members. If you specify both PDS and MEMBERS, Stage 2 **link-edits** the members into the nucleus being generated (IEANUC01).

Figure 3-1 on page 3-15 identifies the system data sets for which you may specify this parameter, and the form that the members must be in for inclusion in the system data sets.

RESIDNT = name
specifies the name(s) of member(s) of a cataloged partitioned data set that Stage 2 copies into the system data set you define. Each name cannot exceed 8 alphameric characters. The first character must be alphabetic. You may specify as many as 10 members; however, the sum of the names you specify using MEMBERS and RESIDNT cannot exceed 20. Stage 2 includes the names you specify in the IEAFIX01 list in SYS1.PARMLIB. At IPL time, MVS includes them in the resident portion of the link pack area. If you specify this parameter, also specify the PDS parameter.

Do not specify the same name in both the RESIDNT parameter and the MEMBERS parameter.

Figure 3-1 on page 3-15 identifies the system data sets for which you may specify this parameter, and the form that the members must be in for inclusion in the system data sets.

SIOAPP = suffix
specifies a user-written start I/O appendage. Specify an alphameric value, from WA through Z9, that is the last 2 characters of the appendage name; Stage 2 automatically assigns the first 6 characters (IGG019) of the name. You may specify as many as 84 appendages. Specify this parameter only for SYS1.LPALIB or SYS1.SVCLIB. If you specify this parameter, also specify the PDS parameter.

SMFDSN = suffix
allows you an alternate method of specifying the VSAM SMF data sets (SYS1.MANn) without specifying the system data set identifier. Using SMFDSN, the suffix(es) you supply replace the n value of SYS1.MANn. Specify the suffix as follows:

```
SMFDSN=n
SMFDSN=(n,n)
SMFDSN=(x-y)
SMFDSN=(x-y,n[,n]...)
```

where either n, x, or y is a value of A through Z or 0 through 9, and x through y is a range of values, where y is greater than x. For instance, SMFDSN = (A-C,1) specifies SYS1.MANA, SYS1.MANB, SYS1.MANC, and SYS1.MAN1. SMFDSN = (A-9) specifies 36 SMF data sets, SYS1.MANA-SYS1.MANZ and SYS1.MAN0-SYS1.MAN9.

SPACE
specifies a request that space allocation be based solely on the values you specify in this parameter.

{**blocksize|CYL|TRK**}
specifies the unit of space you want allocated. You can allocate space in maximum block length, cylinders, or tracks. You must allocate some data sets in specific units. For this information, see the notes to Figure 3-1 on page 3-15.

quantity
specifies how many units of space (blocks, cylinders, or tracks) you want allocated. Depending on the data set, **quantity** may have one of several forms. Generally the form is:

```
(primary quantity[,secondary quantity]
  [,directory blocks])
```

primary quantity
specifies how many units of space you want allocated.

secondary quantity
specifies how many more blocks, cylinders, or tracks are you want allocated if additional space is required. Figure 3-1 identifies the data sets for which you want secondary space allocated. If secondary allocation is not permitted, omit the field; do not code a zero for the secondary quantity.

Caution: An attempt to access a secondary allocation after you extend the data set, but before the next IPL, may cause an abend to occur.

directory blocks
specifies the number of 256-byte blocks you want allocated for the directory of a partitioned data set. Figure 3-1 identifies the system data sets for which this applies.

SWAPDSN = name
specifies the name of a swap data set. The name cannot exceed 44 characters. The characters can be alphameric, @, #, $, or special characters, such as the hyphen. Separate names containing more than 8 characters by means of periods between segments of from 1 to 8 characters; periods count toward the

maximum of 44 characters. The first character of any name or name segment must be an alphabetic or a national character.

MVS does not require swap data sets. You can define swap data sets during (or before) sysgen, but not more than a total of 10 using both this parameter and access method services. (You can define additional swap data sets after sysgen.)

Note: Stage 2 does not append the index value you specify on the GENERATE statement to the swap data set name you specify in this parameter.

For additional information, see Figure 3-1 on page 3-15.

system data set identifier
specifies the name of a system data set you can define during sysgen. The valid names are:

BRODCAST
CMDLIB
DAE
DGTCLIB
DGTLLIB
DGTMLIB
DGTPLIB
DGTSLIB
DGTTLIB
DUMPnn
DUPLEXDS
HELP
IMAGELIB
INDMAC
ISAMLPA
JES3LIB
JES3MAC
LINKLIB
LOGREC
LPALIB
MACLIB
MANn
master catalog (ICFCATLG or VSCATLG)
MIGLIB
MODGEN
NUCLEUS
PARMLIB
PROCLIB
SAMPLIB
SBLSCLI0
SBLSKEL0
SBLSMSG0
SBLSPNL0
SBLSTBL0
STGINDEX
SVCLIB
TCOMMAC
TELCMLIB
UADS
VTAMLIB

VOL

specifies the volume serial number of the volume you want to contain the system data set, page data set, or swap data set, and the generic name of the device on which you want the volume to reside.

{**volser**|*SYSRES*}

specifies the volume serial number. The value cannot exceed 6 alphameric characters.

{**genname**|*3380*}

specifies the generic name of the device. The generic names you may specify are: 2305-2, 3330, 3330-1, 3340, 3350, 3350P, 3351, 3375, 3380, and 3390.

The VOL parameter works in conjunction with the RESVOL parameter on the GENERATE statement. The default values for both are the same. If you specify other than the default values for RESVOL, the default values for VOL change to match the values you specify in the RESVOL parameter.

Example 1: This statement identifies SYS1.LPALIB to the generating system. SYS1.LPALIB has been predefined on a 3380 volume labeled 222222.

```
LPALIB   DATASET   LPALIB,VOL=(222222,3380)
```

Example 2: This statement specifies a VSAM master catalog. The name of the master catalog is SYS1.VSCATLG and it is to reside on the system residence volume.

```
VSCATLG   DATASET   VSCATLG,SPACE=(CYL,(50,5)),              X
            NAME=SYS1.VSCATLG
```

Example 3: This example shows how to specify an integrated catalog facility master catalog. The name of the catalog is SYS1.ICFCATLG and it resides on the system residence volume.

```
ICFCATLG  DATASET  ICFCATLG,SPACE=(CYL,(50,5)),              X
            NAME=SYS1.ICFCATLG
```

Example 4: This statement specifies a page data set, PAGE1, on a 3380 volume labeled PAGE.

```
PAGE1     DATASET  PAGEDSN=PAGE1,VOL=(PAGE,3380),           X
            SPACE=(CYL,(134))
```

Example 5: This statement specifies a swap data set, SWAP5, on a 3380 volume labeled SWAP.

```
SWAP5     DATASET  SWAPDSN=SWAP5,VOL=(SWAP,3380),           X
            SPACE=(CYL,(20))
```

Example 6: This statement specifies a duplex page data set, DUPLEX1, on a 3380 volume labeled DUPLX.

```
DUPLEX1   DATASET  DUPLEXDS,NAME=DUPLEX1,                    X
            VOL=(DUPLX,3380),SPACE=(CYL,(10))
```

Example 7: This statement specifies the SYS1.NUCLEUS data set. Note that no secondary space allocation is allowed in the SPACE parameter.

```
NUCLEUS   DATASET  NUCLEUS,SPACE=(CYL,(30,,10))
```

Example 8: This statement specifies that SYS1.LOGREC is to be allocated on a 3380 volume named 333333. Note that no secondary space allocation is allowed in the SPACE parameter.

```
LOGREC    DATASET  LOGREC,VOL=(333333,3380),SPACE=(CYL,(5))
```

Figure 3-1 (Page 1 of 2). Parameters Supported When Specifying Particular Data Sets with the DATASET Statement

Data Set	Space (unit,	(pri.,	sec.,	dir. blks.))	PDS=	MEMBERS=	RESIDNT=	User-Written Rtns Format	Notes
BRODCAST	x	x							12
CMDLIB	x	x	x	x	x		x	Load Module	2
DAE	x	x	x						3,9,32
DGTCLIB	x	x	x	x	x	x		Clist Library	4,17,33
DGTLLIB	x	x	x	x	x	x		Load Module	4,17,33
DGTMLIB	x	x	x	x	x	x		Message Library	4,17,33
DGTPLIB	x	x	x	x	x	x		Panel Library	4,17,33
DGTSLIB	x	x	x	x	x	x		Skeleton Library	4,17,33
DGTTLIB	x	x	x	x	x	x		Table Library	4,17,33
DUMPnn	x	x							9,11,19, 29,32
DUPLEXDS	x	x							7,8
HELP	x	x	x	x					5,9
IMAGELIB	x	x	x	x	x	x		FCB, UCS Images, 3800 Support Modules	9,10,16, 31
INDMAC	x	x	x	x				Finance Macros	9,23
ISAMLPA	x	x	x	x	x	x		Load Module	12,17,33,34
JES3LIB	x	x	x	x	x	x			17,25,33
JES3MAC	x	x	x	x	x	x		JES3 Macros	12,17,25
LINKLIB	x	x	x	x	x	x	x	Load Module	7,16,17
LOGREC									16,17,32
LPALIB	x	x	x	x	x	x	x	Load Module	12,16,17
MACLIB	x	x	x	x	x	x		System Macros	12,16,17
MANn	x	x							27,31,32
master catalog	x	x	x						8,16,20, 32
MIGLIB	x	x	x	x	x	x	x	Load Module	7,16,17
MODGEN	x	x	x	x	x	x		System Macros	12,16,17
NUCLEUS	x	x		x	x	x		Load Module	7,13,15, 16
page data sets	x	x							6,7,8, 24,33
PARMLIB	x	x		x	x	x		System Parameters	14,16,17
PROCLIB	x	x	x	x	x	x		System Procedures	12,16,17
SAMPLIB	x	x	x	x	x				16,17

Figure 3-1 (Page 2 of 2). Parameters Supported When Specifying Particular Data Sets with the DATASET Statement

Data Set	Space (unit,	(pri.,	sec.,	dir. blks.))	PDS=	MEMBERS=	RESIDNT=	User-Written Rtns Format	Notes
SBLSCLI0	x	x	x	x	x	x		IPCS CLIST	17,33
SBLSKEL0	x	x	x	x	x	x		IPCS Tailoring	17,33
SBLSMSG0	x	x	x	x	x	x		IPCS Dialog Messages	17,33
SBLSPNL0	x	x	x	x	x	x		IPCS Dialog Panels	17,33
SBLSTBL0	x	x	x	x	x	x		IPCS Dialog Table	17,33
STGINDEX	x	x							7,8,21, 28,32
SVCLIB	x	x	x	x	x	x	x	Load Modules	7,13,16, 17
swap data sets	x	x							6,7,8, 33
TCOMMAC	x	x	x	x	x				26
TELCMLIB	x	x	x	x	x	x		Load Modules	9,12,18
UADS	x	x	x	x					1
VTAMLIB	x	x	x	x	x	x		Load Modules	22,33

Notes to Figure 3-1:

Unless otherwise stated, the system data sets listed in Figure 3-1 must reside on a direct access volume, which can be the system residence volume.

1. Catalog this system data set. Allocate space only if you activate TSO.
2. Catalog this system data set and allocate space, unless you specify TSOCMDS=NO on the DATAMGT statement.
3. This system data set is required only if you activate DAE. Specify this data set on a DATASET statement. If you predefine this data set, omit the SPACE parameter from the DATASET statement.
4. This data set is required to use the interactive storage management facility (ISMF).
5. This system data set is required only if you use the TSO HELP command.
6. If you specify the SPACE parameter, also specify the VOL parameter. If you do not specify the SPACE parameter, do not specify the VOL parameter.
7. Allocate space in cylinders.
8. During the first sysgen, allocate space for the master catalog, page data sets, SYS1.STGINDEX, and swap data sets on a single volume. After the initial IPL, performance considerations may require a different configuration.
9. Specify a DATASET statement. If you predefine this system data set, omit the SPACE parameter.

10. Forms control buffer (FCB) modules, universal character set (UCS) images, 3800 character arrangement image sets, 3800 copy modification modules, graphic character modification modules and library character set modules are in this library.
11. This system data set is optional.
12. For improved system efficiency, allocate space in cylinders.
13. This system data set must reside on the system residence volume.
14. If you include a type 3 SVC routine, the name of the SVC routine must be of the form: IGC00*nnn*, where *nnn* is a unique decimal number you assign in descending order from 255 to 200. This name must be the name of a member of a cataloged partitioned data set. If you include a type 4 SVC routine, the name of the SVC routine must be of the form: IGC*ssnnn*, where *nnn* is a unique decimal number you assign in descending order from 255 to 200, and *ss* is a unique decimal number of the load module you assign beginning with 01 for the first (or only) load. This name must be the name of a member of a cataloged partitioned data set. If you specify type 3 or type 4 SVC routines, identify them in an IEASVCxx PARMLIB member or by an SVCUPDTE macro.

    For more information on SVC routines, see IEASVCxx in *Initialization and Tuning*.
15. You can include a maximum of 10 load modules in the nucleus. If you include type 1 or type 2 SVC routines in the nucleus, each load module can contain more than one SVC routine. If you include a type 1 or type 2 SVC routine, the name of the SVC routine must be of the form: IGC*nnn*, where *nnn* is a unique decimal number you assign in descending order from 255 to 200.
16. This system data set is required. You may define it with a DATASET statement.
17. If you do not specify this data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter of the GENERATE statement (the system residence volume).
18. This system data set is required only if you install BTAM/SP and/or ACF/TCAM.
19. Each DASD data set must reside on a single volume.
20. Even if you predefine the master catalog, specify it on a DATASET statement. Define either an integrated catalog facility or a VSAM catalog.
21. Permanently mount or reserve the volume on which this data set resides while it is in use.
22. This system data set is required only if you install ACF/VTAM. Specify it on a DATASET statement.
23. This system data set is required only if you install industry subsystem support.
24. Specify at least 3 data sets. If you predefine the system data sets, omit the SPACE parameter.
25. This system data set is required only if you install JES3.
26. This system data set is required only if you install ACF/TCAM.
27. Allocate space only if SMF or other measurement facilities will do recording. Catalog this data set.
28. If MVS needs to save VIO pages across job steps and between IPLs, this system data set is required.
29. Each DASD data set must reside on a permanently resident volume.

30. This data set may reside on tape.
31. This data set must reside on a permanently mounted volume.
32. Do not share this data set across systems.
33. This data set must be cataloged.
34. This data set is required if ISAM is selected on the DATAMGT macro for the sysgen, or if DFP is installed using the JCLIN provided with the product.

GENERATE

Required

Use the GENERATE statement to specify the volume serial number and generic or esoteric name of the device for the system residence volume of the system being generated, and the output class and job class for use during sysgen. Include this statement just before the assembler END statement in the Stage 1 input.

The serial number you specify in the RESVOL parameter cannot be the serial number of the system residence volume of the generating system. Do not specify the index of the system data sets as SYS1.

[symbol]	GENERATE	[GENTYPE=ALL] [INDEX={name|SYSX}] [JCLASS={class|A}] [OCLASS={class|A}] [RESVOL=([{volser|SYSRES}] [,{genname|3380}])] OBJPDS

GENTYPE = ALL
specifies a sysgen.

INDEX = {**name**|*SYSX*}
specifies the qualifier for the system data sets. The qualifier may be from 1 to 6 alphameric characters; the first character must be alphabetic. Specify an index value other than SYS1; Stage 2 changes the qualifier of the data sets to SYS1.

If you predefine any system data sets, specify their index values the same as the values you specify in this parameter.

This parameter does not apply to the page data sets, the swap data sets, the duplex data set, or the master catalog.

JCLASS = {**class**|*A*}
is a letter from A through O that specifies the job class for Stage 2 jobs.

OCLASS = {**class**|*A*}
is a letter from A through Z or a number from 0 through 9 that specifies the output class for output from Stage 2.

RESVOL

specifies the volume serial number and the generic name of the device on which the system residence volume will reside.

{**volser**|*SYSRES*}

specifies the volume serial number of the system residence volume.

Note: The volume serial number must not be the same as the volume serial number of the system residence pack of the generating system.

{**genname**|*3380*}

specifies the device number or generic name of the device you want used for the system residence volume. Valid generic names are 2305-2, 3330, 3330-1, 3340, 3350, 3375, 3380, and 3390.

The RESVOL parameter works in conjunction with the VOL parameter on the DATASET statement. The default values for both parameters are the same. If you specify other than the default values for RESVOL, the default values for VOL change to match the values you specify in the RESVOL parameter.

OBJPDS

The OBJPDS parameter is no longer supported during sysgen, but is allowed in the Stage 1 deck for compatibility. Maintaining the OBJPDS parameter on the GENERATE statement in your Stage 1 deck causes the following Mnote during Stage 1:

```
0,***IEIGEN103 OBJPDS PARAMETER NO LONGER
     SUPPORTED***
```

Example: This example specifies a sysgen. The system residence volume is a 3380 volume labeled SYSTEM. The index of the system data sets is SG2, which is renamed to SYS1 during Stage 2. The job class and output class are K.

```
GEN   GENERATE   GENTYPE=ALL,RESVOL=(SYSTEM,3380),          X
                 INDEX=SG2,JCLASS=K,OCLASS=K
```

JES

Optional

Starting with the JES3 component of MVS/System Product Version 3, Release 1.3 (FMID HJS3313), the JES statement no longer builds job streams that install JES3. Specifying JES during a sysgen no longer includes JES3 into the system, but it does cause the following Mnote:

```
0,****SYSGEN INSTALLATION OF JES3 IS NO LONGER SUPPORTED****
```

To install JES3, use either the GENERATE command or the RECEIVE, APPLY, and ACCEPT commands. (For details, see the *MVS/ESA Program Directory: JES3* and the *SMP/E User's Guide*.)

Chapter 4. Selecting and Defining the System Data Sets

You can define some or all of the system data sets either during sysgen, by using the DATASET statement, or before sysgen, by using JCL and/or access method services. This chapter contains information about each of the system data sets and how to define them.

"Defining the System Data Sets" tells how to allocate space for system data sets and catalog them in the master catalog. It gives examples of coding the parameters on the DATASET statement and JCL, and the command statements and parameters required to execute access method services.

The examples show the use of JCL and the access method services commands for defining the system data sets.

"System Data Set Summary" on page 4-13 presents an overview of the system data sets that may be helpful, particularly if you define them using JCL and/or access method services. Also in this section are discussions about each of the system data sets, which are arranged in alphabetic order by their qualified names.

Defining the System Data Sets

Before you include components from the DLIBs and user-defined data sets in the system, build the master catalog, allocate space for the system data sets on system volumes, and catalog the data sets in the master catalog.

Using the DATASET Statement to Define the System Data Sets

You can use the DATASET statement to define (allocate space for, and catalog) the system data sets. Include one statement for each data set you define during sysgen. The information necessary for coding the DATASET statement is in Chapter 3, "Coding the Sysgen Statements" on page 3-1. (Chapter 3 also contains information about using the DATASET statement to add user routines to some of the system data sets.) Each of the data sets is discussed later in this chapter. For an example of using the DATASET statement to define the system data sets, see Figure 4-1 on page 4-2.

During Stage 1, the assembler processes the information on the statements and generates the JCL and the commands required to execute access method services.

Stage 2, using the Stage 1 output:

- allocates space on system volumes
- catalogs the system data sets in the master catalog

Stage 2 creates the master catalog first (if you specify it on a DATASET statement). Then, Stage 2 allocates space for the remaining system data sets and catalogs them. Stage 2 also changes the index for all the system data sets to SYS1 from the index name you coded on the GENERATE statement.

```
BRODCAST DATASET  BRODCAST,VOL=(SG2001,3380),SPACE=(CYL,(3))
CMDLIB   DATASET  CMDLIB,VOL=(SG2001,3380),SPACE=(CYL,(6,2,100))
DAE      DATASET  DAE,VOL=(SG2001,3380),SPACE=(TRK,(6,2))
DGTCLIB  DATASET  DGTCLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTLLIB  DATASET  DGTLLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTMLIB  DATASET  DGTMLIB,VOL=(SG2001,3380),SPACE=(CYL,(1,1,20))
DGTPLIB  DATASET  DGTPLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTSLIB  DATASET  DGTSLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTTLIB  DATASET  DGTTLIB,VOL=(SG2001,3380),SPACE=(TRK,(9,1,10))
DUMP00   DATASET  DUMP00,VOL=(SG2001,3380),SPACE=(CYL,(30))
DUMP01   DATASET  DUMP01,VOL=(SG2001,3380),SPACE=(CYL,(30))
HELP     DATASET  HELP,VOL=(SG2001,3380),SPACE=(CYL,(5,1,5))
IMAGELIB DATASET  IMAGELIB,VOL=(SG2001,3380),SPACE=(CYL,(10,3,30))
INDMAC   DATASET  INDMAC,VOL=(SG2001,3380),SPACE=(CYL,(50,3,5))
ISAMLPA  DATASET  ISAMLPA,VOL=(SG2001,3380),SPACE=(CYL,(5,1,30))
JES3LIB  DATASET  JES3LIB,VOL=(SG2001,3380),SPACE=(CYL,(12,1,100))
JES3MAC  DATASET  JES3MAC,VOL=(SG2001,3380),SPACE=(CYL,(15,1,75))
LINKLIB  DATASET  LINKLIB,VOL=(SG2001,3380),SPACE=(CYL,(60,9,600))
LOGREC   DATASET  LOGREC,VOL=(SG2001,3380),SPACE=(CYL,(5))
LPALIB   DATASET  LPALIB,VOL=(SG2001,3380),SPACE=(CYL,(24,3,350))
MACLIB   DATASET  MACLIB,VOL=(SG2001,3380),SPACE=(CYL,(55,5,150))
MIGLIB   DATASET  MIGLIB,VOL=(SG2001,3380),SPACE=(CYL,(60,9,600))
SMFDSNS  DATASET  SMFDSN=(1-9),VOL=(SG2001,3380),SPACE=(CYL,(2))
MODGEN   DATASET  MODGEN,VOL=(SG2001,3380),SPACE=(CYL,(55,5,150))
NUCLEUS  DATASET  NUCLEUS,VOL=(SG2001,3380),SPACE=(CYL,(35,,100))
PARMLIB  DATASET  PARMLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,25))
PROCLIB  DATASET  PROCLIB,VOL=(SG2001,3380),SPACE=(CYL,(3,5,20))
SAMPLIB  DATASET  SAMPLIB,VOL=(SG2001,3380),SPACE=(CYL,(10,5,50))
SBLSCLI0 DATASET  SBLSCLI0,VOL=(SG2001,3380),SPACE=(TRK,(4,4,5))
SBLSKEL0 DATASET  SBLSKEL0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
SBLSMSG0 DATASET  SBLSMSG0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
SBLSPNL0 DATASET  SBLSPNL0,VOL=(SG2001,3380),SPACE=(CYL,(2,1,15))
SBLSTBL0 DATASET  SBLSTBL0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
STGINDEX DATASET  STGINDEX,VOL=(SG2001,3380),SPACE=(CYL,(12))
SVCLIB   DATASET  SVCLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,2,15))
TELCMLIB DATASET  TELCMLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,2,70))
TCOMMAC  DATASET  TCOMMAC,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
UADS     DATASET  UADS,VOL=(SG2001,3380),SPACE=(CYL,(8,1,20))
VTAMLIB  DATASET  VTAMLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,2,50))
ICFCATLG DATASET  ICFCATLG,VOL=(SG2001,3380),SPACE=(CYL,(50,5)),       X
               NAME=ICFCAT
PAGE1    DATASET  PAGEDSN=SG2PAGE1,VOL=(SG2001,3380),SPACE=(CYL,(25))
PAGE2    DATASET  PAGEDSN=SG2PAGE2,VOL=(SG2001,3380),SPACE=(CYL,(25))
PAGE3    DATASET  PAGEDSN=SG2PAGE3,VOL=(SG2001,3380),SPACE=(CYL,(100))
SWAP1    DATASET  SWAPDSN=SG2SWAP1,VOL=(SG2001,3380),SPACE=(CYL,(40))
DUPLEX1  DATASET  DUPLEXDS,NAME=DUPLEX1,VOL=(SG2001,3380),            X
               SPACE=(CYL,(50))
```

Figure 4-1. Using the DATASET Statement to Define the System Data Sets

Using JCL and Access Method Services to Define the System Data Sets

You can use JCL and access method services to predefine the system data sets. Use JCL to allocate space for non-VSAM data sets and access method services to catalog them. Use access method services to catalog and allocate space for VSAM data sets. Generally, the only requirements for predefining the system data sets are:

- Define the system data sets before Stage 2.
- Define the master catalog first, by defining it in the first step of the job and by defining the remaining system data sets in the second step.
- Specify the unique name of the master catalog on a DATASET statement.
- Include a DATASET statement for each system data set that you are predefining, except for those system data sets that must reside on the system residence volume or those that are assumed to reside on it. By omitting the SPACE parameter on a DATASET statement, Stage 2 does not allocate space for a data set.

See Figure 4-2 on page 4-4 through Figure 4-6 on page 4-11 for coding examples that:

1. Use the access method services commands to catalog and allocate space for VSAM system data sets (master catalog, SMF data sets, SYS1.STGINDEX, page and swap data sets, and duplex data sets)
2. Use JCL to allocate space for non-VSAM data sets and the access method services DEFINE command to catalog them

Defining the Master Catalog

You can define either an integrated catalog facility catalog or a VSAM catalog as the master catalog.

Because the generating system already has a master catalog, the system catalogs the catalog you define here (or in job 0 of Stage 2) as a user catalog in the generating system's master catalog. During job 6, the user catalog becomes the master catalog of the generated system.

Defining an integrated catalog facility master catalog

```
//DEFICAT   JOB   MSGLEVEL=1,MSGCLASS=A
//DEFINEIC  EXEC  PGM=IDCAMS
//CATVOL    DD    VOL=SER=SG2001,UNIT=3380,DISP=OLD
//SYSPRINT  DD    SYSOUT=A
//SYSIN     DD    *
  DEFINE  MASTERCATALOG (NAME(ALT1.A.NEW.ICF.CATALOG)    -
                        FILE(CATVOL)                     -
                        ICFCATALOG                       -
                        VOLUME(SG2001)                   -
                        CYLINDERS(5 2))
  LISTCAT   CATALOG (SYS1.A.NEW.ICF.CATALOG)
/*
```

Figure 4-2. Defining an Integrated Catalog Facility Catalog

This job results in the creation of a VSAM volume data set (VVDS), if one does not already exist on the volume. For more information on defining an integrated catalog facility catalog, see *DFP: Planning Guide* and *Catalog Administration Guide*. For information on access method services commands, see *Access Method Services Reference*.

Defining a VSAM master catalog

```
//DEFMCAT   JOB   MSGLEVEL=1,MSGCLASS=A
//DEFINEMC  EXEC  PGM=IDCAMS
//CATVOL    DD    VOL=SER=SG2001,UNIT=3380,DISP=OLD
//SYSPRINT  DD    SYSOUT=A
//SYSIN     DD    *
  DEFINE  MASTERCATALOG (NAME(ALT1.A.NEW.MASTER.CATALOG)   -
                        FILE(CATVOL)                       -
                        VOLUME(SG2001)                     -
                        CYLINDERS(30 5))                   -
          DATA          (CYLINDERS(5 2))
  LISTCAT   CATALOG (SYS1.A.NEW.MASTER.CATALOG)
/*
```

Figure 4-3. Defining a VSAM Master Catalog Using Access Method Services

This job allocates a data space and suballocates from that the master catalog space. The specifications on the MASTERCATALOG and CYLINDERS parameters of the DEFINE command determine the size of the data space allocation (150 cylinders). The specifications on the DATA and CYLINDERS parameters of the DEFINE command determine the size of the space suballocated for the master catalog (10 cylinders).

The VSAM master catalog does not require all the data space; consequently, you can use the remaining space for subsequent allocations of VSAM data sets on the same volume. For more information on defining a VSAM master catalog, see the *Catalog Administration Guide*.

After defining the data space and/or the new integrated catalog facility or VSAM master catalog, use the LISTCAT command to print the names of the entries in the new catalog.

Defining the Non-VSAM System Data Sets

Cataloging the Non-VSAM System Data Sets

Figure 4-4 is an example of how to catalog the non-VSAM system data sets in a master catalog. Notes follow the figure.

```
//CATNVSAM JOB  MSGLEVEL=1,MSGCLASS=A
//NONVSAM  EXEC PGM=IDCAMS
//STEPCAT  DD   DSN=SYS1.A.NEW.ICF.CATALOG,DISP=SHR
//SYSPRINT DD   SYSOUT=A
//SYSIN    DD   *
 DEFINE NONVSAM (NAME(ALT1.BRODCAST) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.CMDLIB)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DAE)      VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTCLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTLLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTMLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTPLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTSLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DGTTLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.DUMP00)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.FDEFLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.FONTLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.HELP)     VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.IMAGELIB) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.INDMAC)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.ISAMLPA)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.JES3LIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.JES3MAC)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.LINKLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.LOGREC)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.LPALIB)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.MACLIB)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.MIGLIB)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.MODGEN)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.NUCLEUS)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.OVERLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.PARMLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.PDEFLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.PROCLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.PSEGLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SAMPLIB)  VOLUMES('******') DEVICETYPES(0000))
 DEFINE NONVSAM (NAME(ALT1.SBLSCLI0) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SBLSKEL0) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SBLSMSG0) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SBLSPNL0) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SBLSTBL0) VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.SVCLIB)   VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.TCOMMAC)  VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.TELCMLIB) VOLUMES(SG2001) DEVICEYTPES(3380))
 DEFINE NONVSAM (NAME(ALT1.UADS)     VOLUMES(SG2001) DEVICETYPES(3380))
 DEFINE NONVSAM (NAME(ALT1.VTAMLIB)  VOLUMES(SG2001) DEVICETYPES(3380))
 LISTCAT LEVEL (ALT1) CATALOG (SYS1.A.NEW.ICF.CATALOG)
/*
```

Figure 4-4. Cataloging the Non-VSAM System Data Sets Using Access Method Services

Notes to Figure 4-4:

The example assumes that you specified ALT1 in the INDEX parameter on the GENERATE statement.

The job CATNVSAM creates catalog entries in the new master catalog for each of the non-VSAM system data sets represented by a DEFINE command.

When the data sets are to reside on the system residence volume, you can specify the VOLUMES and DEVICETYPES parameters using indirect volume serial number and device type, as follows:

```
VOLUMES('******') DEVICETYPES(0000)
```

If you expect to change the system residence volume serial number, this allows you to use the non-VSAM data sets without having to recatalog them to point to the new volume. (For more information on indirect volume serial number and device type, see "DEFINE NONVSAM" in *Access Method Services Reference*.)

Note: For APF-authorized data sets, specify the volume serial number and device type explicitly.

Use the LISTCAT command to print the names of the non-VSAM entries in the new catalog.

Allocating Space for the Non-VSAM System Data Sets

The job ALLOC allocates space for the non-VSAM system data sets that were cataloged by the previous job (CATNVSAM). because these data sets are already cataloged, this job does not access or update the master catalog. For additional information about the DCB subparameters, see Figure 4-5 on page 4-7.

```
//ALLOC    JOB  MSGLEVEL=1,MSGCLASS=A
//ALLOCATE EXEC PGM=IEFBR14
//BRODCAST DD DSN=ALT1.BRODCAST,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            UNIT=3380,SPACE=(CYL,(3),,CONTIG)
//*
//CMDLIB   DD DSN=ALT1.CMDLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(6,2,50))
//*
//DAE      DD DSN=ALT1.DAE,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PS,LRECL=255,BLKSIZE=23205),
//            UNIT=3380,SPACE=(TRK,(6,2))
//*
//DGTCLIB  DD DSN=ALT1.DGTCLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PO,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(CYL,(5,1,50))
//*
//DGTLLIB  DD DSN=ALT1.DGTLLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=U,DSORG=PO,BLKSIZE=6144),
//            UNIT=3380,SPACE=(CYL,(5,1,50))
//*
//DGTMLIB  DD DSN=ALT1.DGTMLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PO,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(CYL,(1,1,20))
//*
//DGTPLIB  DD DSN=ALT1.DGTPLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PO,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(CYL,(5,1,50))
//*
//DGTSLIB  DD DSN=ALT1.DGTSLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PO,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(CYL,(5,1,50))
//*
//DGTTLIB  DD DSN=ALT1.DGTTLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,DSORG=PO,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(TRK,(9,1,10))
//*
//DUMP     DD DSN=ALT1.DUMP00,DISP=(SHR,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=F,BLKSIZE=4160),
//            UNIT=3380,SPACE=(CYL,(12),,CONTIG)
//*
//FDEFLIB  DD DSN=ALT1.FDEFLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=VBM,LRECL=8205,BLKSIZE=8209),
//            UNIT=3380,SPACE=(CYL,(2,1,5))
//*
//FONTLIB  DD DSN=ALT1.FONTLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=VBM,LRECL=8205,BLKSIZE=8209),
//            UNIT=3380,SPACE=(CYL,(12,1,50))
//*
```

Figure 4-5 (Part 1 of 4). Allocating Space for the Non-VSAM System Data Sets Using JCL

```
//HELP     DD DSN=ALT1.HELP,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(5,1,15))
//*
//IMAGELIB DD DSN=ALT1.IMAGELIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(10,3,100))
//*
//INDMAC   DD DSN=ALT1.INDMAC,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            UNIT=3380,DCB=(RECFM=U,BLKSIZE=23440),
//            SPACE=(CYL,(50,3,5)),LABEL=EXPDT=99350
//*
//ISAMLPA  DD DSN=ALT1.ISAMLPA,DISP=(,KEEP),VOL=(,RETAIN, SER=SG2001),
//            UNIT=3380,DCB=(RECFM=U,BLKSIZE=23440),
//            SPACE=(CYL,(5,1,30)),LABEL=EXPDT=99350
//*
//JES3LIB  DD DSN=ALT1.JES3LIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            UNIT=3380,DCB=(RECFM=U,BLKSIZE=23440),
//            SPACE=(CYL,(12,1,100)),LABEL=EXPDT=99350
//*
//JES3MAC  DD DSN=ALT1.JES3MAC,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            UNIT=3380,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            SPACE=(CYL,(15,1,75)),LABEL=EXPDT=99350
//*
//LINKLIB  DD DSN=ALT1.LINKLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(60,9,600))
//*
//LOGREC   DD DSN=ALT1.LOGREC,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,
//            UNIT=3380,SPACE=(CYL,(5),,CONTIG)
//*
//LPALIB   DD DSN=ALT1.LPALIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(24,3,350))
//*
//MACLIB   DD DSN=ALT1.MACLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(55,5,150))
//*
//MIGLIB   DD DSN=ALT1.MIGLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(60,9,600))
//*
//MODGEN   DD DSN=ALT1.MODGEN,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(55,5,150))
//*
//NUCLEUS  DD DSN=ALT1.NUCLEUS,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(35,,100),,CONTIG)
//*
```

Figure 4-5 (Part 2 of 4). Allocating Space for the Non-VSAM System Data Sets Using JCL

```
//OVERLIB  DD DSN=ALT1.OVERLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=VBM,LRECL=8205,BLKSIZE=8209),
//            UNIT=3380,SPACE=(CYL,(2,1,5))
//*
//PARMLIB  DD DSN=ALT1.PARMLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(2,,50),,CONTIG)
//*
//PDEFLIB  DD DSN=ALT1.PDEFLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=VBM,LRECL=8205,BLKSIZE=8209),
//            UNIT=3380,SPACE=(CYL,(2,1,5))
//*
//PROCLIB  DD DSN=ALT1.PROCLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(2,1,50))
//*
//PSEGLIB  DD DSN=ALT1.PSEGLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=VBM,LRECL=8205,BLKSIZE=8209),
//            UNIT=3380,SPACE=(CYL,(2,1,5))
//*
//SAMPLIB  DD DSN=ALT1.SAMPLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=FB,LRECL=80,BLKSIZE=3200),
//            UNIT=3380,SPACE=(CYL,(10,5,50))
//*
//SBLSCLI0 DD DSN=ALT1.SBLSCLI0,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=VB,LRECL=255,BLKSIZE=3360),
//            UNIT=3380,SPACE=(TRK,(4,4,5))
//*
//SBLSKEL0 DD DSN=ALT1.SBLSKEL0,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=12960),
//            UNIT=3380,SPACE=(TRK,(2,1,5))
//*
//SBLSMSG0 DD DSN=ALT1.SBLSMSG0,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(TRK,(2,1,5))
//*
//SBLSPNL0 DD DSN=ALT1.SBLSPNL0,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=3120),
//            UNIT=3380,SPACE=(CYL,(2,1,15))
//*
//SBLSTBL0 DD DSN=ALT1.SBLSTBL0,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(RECFM=FB,LRECL=80,BLKSIZE=12960),
//            UNIT=3380,SPACE=(TRK,(2,1,5))
//*
```

Figure 4-5 (Part 3 of 4). Allocating Space for the Non-VSAM System Data Sets Using JCL

```
//SVCLIB   DD DSN=ALT1.SVCLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(2,1,15))
//*
//TCOMMAC  DD DSN=ALT1.TCOMMAC,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            UNIT=3380,DCB=(RECFM=FB,LRECL=80,BLKSIZE=23440),
//            SPACE=(CYL,(5,1,50)),LABEL=EXPDT=99350
//*
//TELCMLIB DD DSN=ALT1.TELCMLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(4,1,100))
//*
//UADS     DD DSN=ALT1.UADS,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            DCB=(DSORG=PO,LRECL=80,RECFM=FB,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(5,2,70))
/*
//VTAMLIB  DD DSN=ALT1.VTAMLIB,DISP=(,KEEP),VOL=(,RETAIN,SER=SG2001),
//            LABEL=EXPDT=99350,DCB=(RECFM=U,BLKSIZE=23440),
//            UNIT=3380,SPACE=(CYL,(5,2,50))
/*
```

Figure 4-5 (Part 4 of 4). Allocating Space for the Non-VSAM System Data Sets Using JCL

Note to Figure 4-5:

The example assumes that you specified ALT1 in the INDEX parameter on the GENERATE statement.

Defining the VSAM System Data Sets

Cataloging and Allocating Space for the VSAM System Data Sets

The job DEFVSAM (see Figure 4-6) defines the VSAM system data sets—the storage index, the page data sets, the swap data set(s), a duplex data set, and an SMF data set.

```
//DEFVSAM  JOB  MSGLEVEL=1,MSGCLASS=1
//VSAMDEF  EXEC PGM=IDCAMS
//STEPCAT  DD   DSN=SYS1.A.NEW.ICF.CATALOG,DISP=OLD
//VOLUME   DD   DISP=OLD,VOL=SER=SG2001,UNIT=3380
//SYSABEND DD   SYSOUT=A
//SYSPRINT DD   SYSOUT=A
//SYSIN    DD   *
  DEFINE CLUSTER          (NAME (ALT1.STGINDEX)       -
                          VOLUMES (SG2001)            -
                          CYLINDERS (5)               -
                          KEYS (12 8)                 -
                          BUFFERSPACE (20480)         -
                          RECORDSIZE (2041 2041)      -
                          FILE (VOLUME)               -
                          REUSE)                      -
  DATA                    (CONTROLINTERVALSIZE (2048))-
  INDEX                   (CONTROLINTERVALSIZE (4096))
  DEFINE CLUSTER          (NAME (ALT1.MAN1)           -
                          VOLUMES (SG2001)            -
                          NONINDEXED                  -
                          CYLINDERS (12)              -
                          REUSE                       -
                          RECORDSIZE (4086 32767)     -
                          SPEED                       -
                          SPANNED                     -
                          CONTROLINTERVALSIZE (4096)  -
                          SHAREOPTIONS (2))
  DEFINE PAGESPACE        (NAME (SYS1.PAGE1)          -
                          CYLINDERS (15)              -
                          FILE (VOLUME)               -
                          VOLUMES (SG2001))
  DEFINE PAGESPACE        (NAME (SYS1.PAGE2)          -
                          CYLINDERS (75)              -
                          FILE (VOLUME)               -
                          VOLUMES (SG2001))
  DEFINE PAGESPACE        (NAME (SYS1.PAGE3)          -
                          CYLINDERS (100)             -
                          FILE (VOLUME)               -
                          VOLUMES (SG2001))
```

Figure 4-6 (Part 1 of 2). Defining the VSAM System Data Sets, Using Access Method Services

```
  DEFINE PAGESPACE        (NAME (SYS1.SWAP1)          -
                          CYLINDERS (20)              -
                          FILE (VOLUME)               -
                          VOLUMES (SG2001)            -
                          SWAP)
  DEFINE PAGESPACE        (NAME (SYS1.DUPLEX)         -
                          CYLINDERS (25)              -
                          FILE (VOLUME)               -
                          VOLUMES (SG2001))
  LISTCAT ALL

/*
```

Figure 4-6 (Part 2 of 2). Defining the VSAM System Data Sets, Using Access Method Services

Note to Figure 4-6:

The example assumes that you specified ALT1 in the INDEX parameter on the GENERATE statement. The first DEFINE CLUSTER command defines a VSAM key-sequenced data set with the characteristics of the system data set SYS1.STGINDEX.

The second DEFINE CLUSTER command defines a VSAM entry-sequenced data set with the characteristics of an SMF system data set.

The first three DEFINE PAGESPACE commands define system paging data sets for PLPA, COMMON, and the first LOCAL, respectively. You must define at least 3 page data sets. The total space for the first (PLPA) and second (COMMON) page data sets must be large enough to hold the PLPA (pageable link pack area) and CSA (common system area) pages. The remaining page data set, LOCAL, must be large enough to hold VIO pages and pageable private area pages.

If you do not define any swap data sets, MVS uses the local page data sets to store the LSQA (local system queue area) areas of address spaces swapped out.

The fourth DEFINE PAGESPACE command defines a swap data set. You can define more than one swap data set. Allocate enough space to hold LSQA areas of address spaces swapped out.

The last DEFINE PAGESPACE command defines an optional page data set, the duplex data set. (If you define a duplex data set, still define at least three other page data sets.) Because duplex processing is done only for PLPA and CSA pages, the duplex data set need never be larger than the combined space of the first (PLPA) and second (COMMON) page data sets you define.

For information on how to determine the amount of space to allocate for each page or swap data set, see *Initialization and Tuning*.

For the initial sysgen, the example defines all data sets on one volume; however, after the first IPL, distribute these data sets on separate volumes for improved performance.

Finally, the LISTCAT command lists each entry in the catalog and its attributes.

System Data Set Summary

Figure 4-7 lists the required and optional system data sets. This figure may be helpful to refer to when you are defining the system data sets. If you allocate space for the non-VSAM system data sets before Stage 2 using JCL, specify the values shown for the DCB subparameters. (You cannot specify DCB subparameter values in the DATASET statement.)

Figure 4-7 (Page 1 of 2). Summary of Required and Optional System Data Sets

System Data Set	Type	Required or Optional Data Set	Is Secondary Space Allocation Allowed?	DCB Parameters	Notes
Master Catalog	VSAM	required	yes	none	1,10,15
BRODCAST	BDAM	optional	no	none	9
CMDLIB	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn	2,13
DAE	seq.	optional	yes	DSORG=PS,RECFM=FB, LRECL=255,BLKSIZE=nnnnn	10
DGTCLIB	PDS	optional	yes	RECFM=FB,BLKSIZE=nnnnn, LRECL=80	3,27
DGTLLIB	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn,	2,27
DGTMLIB	PDS	optional	yes	RECFM=FB,BLKSIZE=nnnnn, LRECL=80	3,27
DGTPLIB	PDS	optional	yes	RECFM=FB,BLKSIZE=nnnnn, LRECL=80	3,27
DGTSLIB	PDS	optional	yes	RECFM=FB,BLKSIZE=nnnnn, LRECL=80	3,27
DGTTLIB	PDS	optional	yes	RECFM=FB,BLKSIZE=nnnnn, LRECL=80	3,27
DUMPnn	seq.	optional	no	RECFM=F,BLKSIZE=4160	6,10,14, 21,22
Duplex Data Sets	VSAM	optional	no	none	11,15
FDEFLIB	PDS	optional	yes	RECFM=VBM,LRECL=8205, BLKSIZE=8209	25,26
FONTLIB	PDS	optional	yes	RECFM=VBM,LRECL=8205, BLKSIZE=8209	25,26
HELP	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,7,14
IMAGELIB	PDS	required	yes	RECFM=U,BLKSIZE=nnnnn	2,14,22
INDMAC	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8,14,18
ISAMLPA	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn	8,28
JES3LIB	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn	2,4
JES3MAC	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,4,8
LINKLIB	PDS	required	yes	RECFM=U,BLKSIZE=nnnnn	2,8
LOGREC	seq.	required	no	none	10
LPALIB	PDS	required	yes	RECFM=U,BLKSIZE=nnnnn	2,8
MACLIB	PDS	required	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8
MANn	VSAM	optional	no	none	10,20,22
MIGLIB	PDS	required	yes	RECFM=U,BLKSIZE=nnnnn	8

Figure 4-7 (Page 2 of 2). Summary of Required and Optional System Data Sets

System Data Set	Type	Required or Optional Data Set	Is Secondary Space Allocation Allowed?	DCB Parameters	Notes
MODGEN	PDS	required	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8
NUCLEUS	PDS	required	no	RECFM=U,BLKSIZE=nnnnn	2,5,8
OVERLIB	PDS	optional	yes	RECFM=VBM,LRECL=8205, BLKSIZE=8209	25,26
Page Data Sets	VSAM	required	yes	none	11,15
PARMLIB	PDS	required	no	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8
PDEFLIB	PDS	optional	yes	RECFM=VBM,LRECL=8205, BLKSIZE=8209	25,26
PROCLIB	PDS	required	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8
PSEGLIB	PDS	optional	yes	RECFM=VBM,LRECL=8205, BLKSIZE=8209	25,26
SAMPLIB	PDS	required	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	24
SBLSCLI0	PDS	optional	yes	RECFM=VB,LRECL=255, BLKSIZE=3360	13
SBLSKEL0	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,13
SBLSMSG0	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,13
SBLSPNL0	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,13
SBLSTBL0	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,13
STGINDEX	VSAM	optional	no	none	10,11,15, 16
SVCLIB	PDS	required	yes	RECFM=U,BLKSIZE=nnnnn	2,5,8
Swap Data Sets	VSAM	optional	no	none	11,15
TCOMMAC	PDS	optional	yes	RECFM=FB,LRECL=80, BLKSIZE=nnnnn	3,8,19
TELCMLIB	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn	2,8,12,14
UADS	PDS	optional	yes	DSORG=PO,RECFM=FB, LRECL=nn,BLKSIZE=nnnnn	9,23
VTAMLIB	PDS	optional	yes	RECFM=U,BLKSIZE=nnnnn	2,17

Notes to Figure 4-7:

1. Even if you predefine the master catalog, specify it on a DATASET statement. Define either an integrated catalog facility or a VSAM master catalog.
2. Use BLKSIZE=14660 for a 2305 Model 2, BLKSIZE=13030 for a 3330 or 3330 Model 11, BLKSIZE=8368 for a 3340, BLKSIZE=18432 for a 3350, BLKSIZE=17600 for a 3375, BLKSIZE=18432 for a 3380, and BLKSIZE=18432 for a 3390.
3. The value of BLKSIZE must be a multiple of 80 that is less than or equal to 14640 for a 2305 Model 2, 12960 for a 3330 or 3330 Model 11, 8320 for a 3340, 19040 for a 3350, 17440 for a 3375, 23440 for a 3380, or 27920 for a 3390.

4. This system data set is required only if you install JES3.
5. Store this system data set on the system residence volume.
6. You may specify from 1 to 10 DUMP*nn* (DUMP00—DUMP09) system data sets.
7. This system data set is required only if you use the TSO HELP command.
8. For improved system efficiency, allocate space in cylinders.
9. Catalog this system data set. Allocate space only if you activate TSO.
10. Do not share this system data set across systems.
11. Allocate space in cylinders.
12. This system data set is required only if you install BTAM/SP and/or ACF/TCAM.
13. Catalog this system data set and allocate space, unless you specify TSOCMDS=NO on the DATAMGT statement.
14. Even if you predefine this system data set, specify it on a DATASET statement and omit the space parameter.
15. During the first sysgen, allocate space for the master catalog, page data sets, swap data sets, and SYS1.STGINDEX on a single volume. After the initial IPL, performance considerations may require a different configuration.
16. The STGINDEX data set must reside on a volume that is permanently mounted or reserved while in use.
17. This system data set is required only if you install ACF/VTAM.
18. This system data set is required only if you install the industry subsystem support.
19. This system data set is required only if you install ACF/TCAM. Specify it on a DATASET statement.
20. Allocate space only if SMF or other measurement facilities will do recording. Catalog this data set.
21. Each DASD data set must reside on a single volume.
22. Each DASD data set must reside on a permanently resident volume.
23. The value of LRECL must be a multiple of 80 for TSO, and 172 for TSO/E.
24. Because SAMPLIB contains object module input to the linkage editor, the BLKSIZE must be a multiple of 80, but not greater than 3200.
25. Define this library as a partitioned data set with a record format of VBM, VBA, VM, or VA.
26. This system data set is required only if you install Print Services Facility (PSF).
27. This system data set is required to use the interactive storage management facility (ISMF).
28. This data set is required if ISAM is selected on the DATAMGT macro for the sysgen, or if DFP is installed using the JCLIN provided with the product.

Master Catalog

Contents: The master catalog contains data set and volume information necessary to locate data sets and user catalogs.

Requirements for Specification

General Information: The master catalog is required. It must reside on a direct access volume, which can be the system residence volume.

Define the master catalog as an integrated catalog facility catalog, although you can define it as a VSAM catalog. For information on integrated catalog facility catalogs, see *Catalog Administration Guide*. Assign the catalog a unique name, and specify the name on a DATASET statement (even if you predefine the master catalog).

To provide faster response to concurrent requests for catalog entries, you can have the VSAM master catalog primarily contain pointers to user catalogs that contain entries for most data sets, indexes, and volumes.

Using DATASET: Specify the NAME parameter. Allocate space only in units of tracks or cylinders.

Using Access Method Services: Define the master catalog as the first system data set. Specify a DATASET statement with the NAME parameter, even if you predefine the master catalog.

Note: Password protect the master catalog so that you can password protect the VSAM data sets cataloged in it. You can protect the master catalog by specifying the password and integrity parameters. Update the master catalog with passwords *after* Stage 2 completes to avoid password requests during Stage 2.

After IPLing the newly generated system, you can use the access method services ALTER command to password protect the new master catalog:

```
//ALTPSWD     JOB   ACCT123,PROGRAMMER,MSGLEVEL=(1,1)
//STEP1       EXEC  PGM=IDCAMS
//SYSPRINT    DD    SYSOUT=A
//SYSIN       DD    *
    ALTER -
         SYS1.A.NEW.ICF.CATALOG -
         MASTERPW(masterpassword) -
         CONTROLPW(controlpassword)
/*
```

You may want to use the master catalog from a previous production system. Chapter 7, "Using Data Sets from the Old Production System" on page 7-1 contains an example and a procedure for combining the system data sets from a newly generated system with the master catalog, SYS1.STGINDEX, and page data sets from an existing production system.

Note: Do not share the master catalog across systems because a problem in the master catalog might interfere with recovery.

Additional Information: Information about using access method services to define the master catalog can be found in *Access Method Services Reference*. For information about the master catalog, see *Catalog Administration Guide*.

SYS1.BRODCAST

Contents: SYS1.BRODCAST is a BDAM data set that contains two types of TSO messages:

- Notices—messages available to all users of a system
- Mail—messages available to specific users of a system

This system data set also contains a notice directory to facilitate the accessing of each type of message.

Requirements for Specification

General Information: SYS1.BRODCAST must reside on a direct access volume, which can be the system residence volume. Catalog it; but allocate space only if you activate TSO. Do not allocate secondary space.

Using JCL and Access Method Services: Do not specify any DCB subparameters. Use either access method services or JCL for cataloging.

Notes:

1. Do not specify an expiration date for this data set.
2. Do not share this data set across systems unless GRS is installed and active.

SYS1.CMDLIB

Contents: SYS1.CMDLIB is a partitioned data set that contains TSO command processor routines, service routines, and utility programs.

Requirements for Specification

General Information: SYS1.CMDLIB must reside on a direct access volume, which can be the system residence volume.

Catalog it and allocate space unless you specify TSOCMDS=NO on the DATAMGT statement. You may allocate secondary space.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.DAE

Contents: SYS1.DAE is a fixed-block data set that contains a permanent record of unique dumps identified by DAE.

Requirements for Specification

General Information: SYS1.DAE is required only if you activate DAE. This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set and either password protect or RACF protect it. You may allocate secondary space.

When using SYS1.DAE, specify DISP=SHR so that you can browse records of unique dumps. You can add, update, or delete records in SYS1.DAE using TSO or ISPF editors or other data set facilities, but do not edit the symptom string.

Using the DATASET statement: If you activate DAE, specify this data set on a DATASET statement, even if you predefine it. If you predefine it, omit the SPACE parameter.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=FB,

LRECL=255,

DSORG=PS,

BLKSIZE=nnnnn

where nnnnn is:

```
14535 for a 2305 Model 2
13005 for a 3330 Model 1, 2, or 11
13005 for a 3333 Model 1 or 11
 8160 for a 3340/3344
18870 for a 3350
17595 for a 3375
23205 for a 3380
```

The preceding values represent optimal space utilization for the above devices.

Note: Do not share this data set across systems.

Additional Information: For additional information on DAE and the use of this data set, see *Planning: Dump and Trace Services*.

SYS1.DGTCLIB

Contents: SYS1.DGTCLIB is a partitioned data set that contains the CLIST text for the interactive storage management facility (ISMF). This text requires the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

This data set requires Data Facility Product (DFP) 3.1.0 or higher.

When using ISMF, you must also allocate this data set to your TSO session.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is a CLIST data set, concatenate it to other CLIST data sets. Specify the same DCB subparameters for this data set as you do for the other CLIST data sets.

If you have not yet established CLIST data set conventions, specify a nominal set of DCB subparameters, such as:

RECFM=VB
LRECL=255
BLKSIZE=3360

Note: To avoid overwriting ISPF, you must concatenate this data set in front of the ISPF CLIST library.

SYS1.DGTLLIB

Contents: SYS1.DGTLLIB is a partitioned data set that contains system load modules for the interactive storage management facility (ISMF). These modules require the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is an ISPF data set and must be concatenated to other ISPF data sets, it is recommended that the DCB subparameters used by the location for other ISPF data sets also be used for this data set.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=U
BLKSIZE=nnnnn

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: To avoid overwriting ISPF, you must concatenate this data set in front of the ISPF load library.

SYS1.DGTMLIB

Contents: SYS1.DGTMLIB is a partitioned data set that contains the message text for the interactive storage management facility (ISMF). These messages require the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is an ISPF data set, and must be concatenated to other ISPF data sets, it is recommended that the DCB subparameters used by the location for other ISPF data sets also be used for this data set.

If ISPF is not yet installed, a nominal set of DCB subparameters can be specified, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: This data set must be concatenated in front of the ISPF message library.

SYS1.DGTPLIB

Contents: SYS1.DGTPLIB is a partitioned data set that contains the panels for the interactive storage management facility (ISMF). These panels require the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is an ISPF data set, and must be concatenated to other ISPF data sets, it is recommended that the DCB subparameters used by the location for other ISPF data sets also be used for this data set.

If ISPF is not yet installed, a nominal set of DCB subparameters can be specified, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: This data set must be concatenated in front of the ISPF panel library.

SYS1.DGTSLIB

Contents: SYS1.DGTSLIB is a partitioned data set that contains the skeletons for the interactive storage management facility (ISMF). These skeletons require the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

This data set requires Data Facility Product (DFP) 3.1.0 or higher.

When using ISMF, you must also allocate this data set to your TSO session.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is an ISPF data set, and must be concatenated to other ISPF data sets, it is recommended that the DCB subparameters used by the location for other ISPF data sets also be used for this data set.

If ISPF is not yet installed, a nominal set of DCB subparameters can be specified, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: This data set must be concatenated in front of the ISPF panel library.

SYS1.DGTTLIB

Contents: SYS1.DGTTLIB is a partitioned data set that contains the tables for the interactive storage management facility (ISMF). These tables require the installation of the Interactive System Productivity Facility (ISPF).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. It must be cataloged, and secondary space may be allocated.

Using the DATASET Macro: If this system data set is not specified in a DATASET macro, it will reside on the volume specified in the RESVOL parameter of the GENERATE macro (the system residence volume).

Using JCL and Access Method Services: Either may be used for cataloging.

Because this data set is an ISPF data set, and must be concatenated to other ISPF data sets, it is recommended that the DCB subparameters used by the location for other ISPF data sets also be used for this data set.

If ISPF is not yet installed, a nominal set of DCB subparameters can be specified, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: This data set must be concatenated in front of the ISPF table library.

SYS1.DUMPnn

Contents: The SYS1.DUMPnn system data sets (SYS1.DUMP00 through SYS1.DUMP09) are sequential data sets that contain system dumps, which record areas of virtual storage in case of system task failures.

Note: Although there can be as many as 100 SYS1.DUMPnn data sets (SYS1.DUMP00 through SYS1.DUMP99), the system generation process supports only ten (SYS1.DUMP00 through SYS1.DUMP09).

Requirements for Specification

General Information: SYS1.DUMP data sets are optional. You can define from 1 to 10 system dump data sets. They all can reside on tape, all on direct access devices, or on a combination of devices. You can add additional dump data sets after IPL with the DUMPDS command.

DASD dump data sets:

- You can define dump data sets before, during, or after sysgen.
- Dump data sets can reside on the system residence volume.
- Each dump data set must reside on only one volume; each dump date set will contain only one dump.
- Specifications:
  - Allocate space for, and catalog, the dump data sets.
  - Place dump data sets on permanently resident volumes.
  - Do not allocate secondary space.
  - Specify CONTIG in the SPACE parameter.
  - Allocate several dump data sets that are large enough to contain the maximum size of any expected SVC dump.

Tape dump data sets:

You can define dump data sets that are to reside on tape during IPL.

Using the DATASET Statement: If you predefine a dump data set with an index other than SYS1, specify it on a DATASET statement; rename the data set to SYS1.DUMPnn; and omit the SPACE parameter.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=F,BLKSIZE=4160

DISP=(SHR,KEEP)
```

Note: Do not share the dump data sets across systems.

Additional Information: For further information see *Planning: Dump and Trace Services*.

SYS1.FDEFLIB

Contents: SYS1.FDEFLIB, a partitioned data set, is required only if install the Print Management Facility with a printer that it can drive, such as the IBM 3800 Model 3 Printing Subsystem.

Requirements for Specification

General Information: SYS1.FDEFLIB is required only if you install the Print Services Facility (PSF) with a printer that can be driven by PSF, such as the IBM 3800 Model 3 Printing Subsystem. This system data set must reside on a direct access volume, which can be the system residence volume. When using SYS1.FDEFLIB, specify DISP=SHR.

Using JCL and Access Method Services: Allocate space for, and catalog, this data set. Use either access method services or JCL for cataloging.

Sample source objects are shipped with PSF. You can build these source objects into SYS1.FDEFLIB using PSF.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=VBM

LRECL=8205

BLKSIZE=8209

SYS1.FONTLIB

Contents: SYS1.FONTLIB is a partitioned data set that contains various font objects (coded fonts, font character sets and code pages built by the Print Management Facility (PMF)).

Requirements for Specification

General Information: SYS1.FONTLIB is required only if you install the Print Services Facility (PSF) with a printer that can be driven by PSF, such as the IBM 3800 Model 3 Printing Subsystem. This system data set must reside on a direct access volume, which can be the system residence volume. When using SYS1.FONTLIB, specify DISP=SHR.

Using JCL and Access Method Services: Allocate space for, and catalog, this data set. Use either access method services or JCL for cataloging.

You can copy the font objects shipped with PSF, which reside in the DLIBs, into SYS1.FONTLIB using the existing ACCEPT or APPLY commands of SMP, or with the IEBCOPY utility.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=VBM

LRECL=8205

BLKSIZE=8209

SYS1.HELP

Contents: SYS1.HELP is a partitioned data set that contains HELP information regarding the syntax, operands, and function of each TSO command.

Requirements for Specification

General Information: SYS1.HELP is required only if you use the TSO HELP command. This system data set must reside on a direct access volume, which can be the system residence volume. You may allocate secondary space.

Using the DATASET Statement: If you use the TSO HELP command, specify this data set on a DATASET statement, even if you predefine it. If you predefine it, omit the SPACE parameter.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Additional Information: For information on this system data set, see *TSO/E Command Reference.*

SYS1.IMAGELIB

Contents: SYS1.IMAGELIB is a partitioned data set that contains the 1403, 3203 Model 5, and 3211 universal character set (UCS) images; 3203 Model 5, 3211, and 3800 forms control buffer (FCB) modules; 3800 character arrangement table modules; 3800 graphic character modification modules; 3800 copy modification modules; 3800 library character sets; 3525 data protection images (DPI); and 3890 document processor SCI programs.

Utilities describes the use of the IEBIMAGE utility to install the IBM-supplied library character sets in SYS1.IMAGELIB. You may prefer to install library character sets in a different library before sysgen, and use the MEMBERS and PDS parameters on the DATASET statement to copy these library character sets from your library to SYS1.IMAGELIB during sysgen.

For information on adding either an IBM UCS (universal character set) or an IBM FCB (forms control buffer) image to SYS1.IMAGELIB, see *System—Data Administration*.

Requirements for Specification

General Information: SYS1.IMAGELIB is required. This system data set must reside on a permanently mounted direct access volume, which can be the system residence volume. You may allocate secondary space.

Using the DATASET Statement: Specify this system data set on a DATASET statement, even if you predefine it. If you predefine it, omit the SPACE parameter.

You can include in this data set user-written forms control buffer (FCB) images, universal character set (UCS) images, and 3800 support modules.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: Installations sensitive to security considerations should protect SYS1.IMAGELIB to ensure that unauthorized programs cannot gain authorization by link-editing into this library.

Additional Information: For more information on maintaining SYS1.IMAGELIB, see the IEBIMAGE program in *Utilities*.

SYS1.INDMAC

Contents: SYS1.INDMAC is a partitioned data set that contains the macro definitions for the industry subsystems.

Requirements for Specification

General Information: SYS1.INDMAC is required only if you install industry subsystem support. This system data set must reside on a direct access volume, which can be the system residence volume. You may allocate secondary space.

Using the DATASET Statement: If you install industry subsystem support, specify this data set on a DATASET statement, even if you predefine it. If you predefine it, omit the SPACE parameter.

You can include user-written macros in SYS1.INDMAC during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.ISAMLPA

Contents: SYS1.ISAMLPA is a partitioned data set that contains all the ISAM component modules loaded into the pageable link pack area (PLPA).

Requirements for Specification

General Information: This system data set must reside on a direct access volume, which can be the system residence volume. Allocate space on a cylinder boundary. You may allocate secondary space.

MVS loads all the modules in this system data set into the LPA at the IPL time if the data set is specified in the LPA list concatenation; therefore, you may place it on a demountable volume that you can remove after IPL.

Using the DATASET Statement: If you do not specify this system data set on a data set statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
19040 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
32760 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.JES3LIB

Contents: SYS1.JES3LIB is a partitioned data set that contains all the JES3 code except the modules that reside in SYS1.LPALIB and SYS1.LINKLIB.

Requirements for Specification

General Information: This system data set is required only if you install JES3. This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.JES3MAC

Contents: SYS1.JES3MAC is a partitioned data set that contains the macro definitions for JES3.

Requirements for Specification

General Information: This system data set is required only if you install JES3. This system data set must reside on a direct access volume, which can be the system residence volume.

For improved system efficiency, allocate space in cylinders. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written system macros in SYS1.JES3MAC during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.LINKLIB

Contents: SYS1.LINKLIB is a partitioned data set that contains programs and routines referred to by the XCTL, ATTACH, LINK, and LOAD macro instructions and nonresident system routines. SYS1.LINKLIB also contains the assembler program, the linkage editor, the utility programs, and service aids.

Requirements for Specification

General Information: SYS1.LINKLIB is required. This system data set must reside on a direct access volume, which can be the system residence volume.

Allocate space in cylinders. For maximum efficiency, do not use alternate tracks. You may allocate secondary space.

You should allocate LNKLST data sets exclusively with primary extents; otherwise, updates to LNKLST libraries could cause the data set to expand into secondary extents. See *Initialization and Tuning*.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written routines, in load module form, in SYS1.LINKLIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.LOGREC

Contents: SYS1.LOGREC is a sequential data set that contains statistical data about machine failures (processor failures, I/O device errors, channel errors). It also contains records for program error recording, missing-interrupt information, and dynamic device reconfiguration (DDR) routines.

General Information: SYS1.LOGREC is required. This system data set must reside on a direct access volume, which may be the system residence volume. Do not allocate secondary space.

IFCDIP00 initializes this data set during Stage 2.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, pre-allocate it on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

Using JCL and Access Method Services: Do not specify DCB subparameters. Use either JCL or access method services for cataloging.

Notes:

1. Do not share this data set across systems.
2. After IPLing the newly generated system, execute the IFCDIP00 program to reinitialize SYS1.LOGREC.

Additional Information: You can increase or decrease the size of SYS1.LOGREC after sysgen using the IFCDIP00 program. Information on this program is given in *SYS1.LOGREC Error Recording*.

SYS1.LPALIB

Contents: SYS1.LPALIB is a partitioned data set that contains all the modules loaded into the pageable link pack area (PLPA). This includes system routines, SVC routines, data management access methods, nonresident machine-check handler modules, authorization and accounting exit routines, logon mode tables, and some TSO modules.

Requirements for Specification

General Information: SYS1.LPALIB is required. This system data set must reside on a direct access volume, which can be the system residence volume. Allocate space on a cylinder boundary. You may allocate secondary space.

MVS loads all the modules in this system data set into the LPA at IPL time; therefore, you may place it on a demountable volume that you can remove after IPL.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written routines, in load module form, in SYS1.LPALIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, Specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.MACLIB

Contents: SYS1.MACLIB is a partitioned data set that contains the macro definitions for supervisor and data management macro instructions.

Requirements for Specification

General Information: SYS1.MACLIB is required. This system data set must reside on a direct access volume, which can be the system residence volume.

For improved system efficiency, allocate space in cylinders. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written system macros in SYS1.MACLIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.MANn

Contents: The SYS1.MANn system data sets (SYS1.MANA through SYS1.MANZ and SYS1.MAN0 through SYS1.MAN9) are VSAM data sets that contain information collected by the system management facilities (SMF) routines or other measurement facilities.

Requirements for Specification

General Information: SYS1.MANn are required only if SMF or other measurement facilities will do recording. They must reside on a permanently mounted direct access volume, which can be the system residence volume. Do not allocate secondary space.

You can define these system data sets at any time.

Using the DATASET Statement: Allocate space only if SMF or other measurement facilities will do recording.

Using JCL and Access Method Services: You can use access method services to define these data sets. To define SYS1.MANn data sets, specify the following command and parameters with appropriate values where indicated:

```
DEFINE CLUSTER -
 (NAME(SYS1.MANn) -
 VOLUME(volser) -
 FILE(ddname) -
 NONINDEXED -
 CYLINDERS(numcyl) -
 REUSE -
 RECORDSIZE(4086,32767) -
 SPEED -
 SPANNED -
 CONTROLINTERVALSIZE(4096) -
 SHAREOPTIONS(2)) -
```

Notes:

1. Do not share these data sets across systems.
2. Values shown in this example are sample values.

Selecting the SMF Data Set Control Interval: With certain restrictions, the control-interval (CI) size of SMF data sets can range from 0.5K (512 bytes) to 22K (22528 bytes). You specify the control-interval size of the SMF data set and the device type when you define the VSAM data set. After you specify the control-interval size, VSAM chooses the physical record size of the data set based on this control-interval size and the track size of the device type.

Valid examples of user CI-size choices (in kilobytes), and physical-record sizes then chosen by VSAM:

Control Interval Size	Physical Record Size
4K (4096 bytes)	4K (4096 bytes)
22K (22528 bytes)	22K (22528 bytes)

The optimal control-interval size is 16K (16384 bytes) for the 3350 direct-access storage device, 22K (22528 bytes) for the 3380 direct-access storage device, and 18K (18432 bytes) for the 3390 direct-access storage device. If you specify an incorrect control-interval size (such as 32K), VSAM chooses an unequal (and thereby incorrect) physical record size (such as 8K).

If the physical record size is not equal to the control-interval size, SMF cannot open the data set. Instead, in addition to displaying the 'feedback' code, SMF issues an error message to the console. If SMF encounters a data set with an incorrect control-interval size, SMF cannot use that data set. The console displays a message informing the operator about the problem. A data set that has any error does not go on the list of active data sets. If all data sets fail to open and allocate successfully, SMF chooses 4K (4096 bytes) as the control-interval size for the initial program load (IPL), and SMF buffers the data.

Additional Information: For information about SMF, see *SMF*, in particular, "SMF Data Sets".

SYS1.MIGLIB

Contents: SYS1.MIGLIB is the system load library for the interactive problem control system (IPCS) and all component and subsystem dump exit modules. The SYS1.MIGLIB library replaces the migration procedures associated with SYS1.SAMPLIB.

Requirements for Specification

General Information: SYS1.MIGLIB is a required system load library. This system data set must reside on a direct access volume, which can be the system residence volume.

Note: SYS1.MIGLIB is automatically included in the LNKLST concatenation, which is formed by concatenating programming libraries to SYS1.LINKLIB. The LNKLSTxx member of SYS1.LINKLIB contains the names of the programming libraries that are to be concatenated.

The component and subsystem dump exits that must function during SNAP processing must also reside in SYS1.LPALIB.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
32760 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

Using JCL and Access Method Services: Use either for cataloging.

SYS1.MODGEN

Contents: SYS1.MODGEN is a partitioned data set that contains the macro definitions for supervisor and data management macro instructions.

Requirements for Specification

General Information: SYS1.MODGEN is required. This system data set must reside on a direct access volume, which can be the system residence volume.

For improved system efficiency, allocate space in cylinders. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written system macros in SYS1.MODGEN during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.NUCLEUS

Contents: SYS1.NUCLEUS is a partitioned data set that contains the resident portion of the control program in two members, IEAVEDAT (DAT-off) and IEANUC0x (DAT-on). SYS1.NUCLEUS also contains the nucleus initialization program (NIP), a pointer to the master catalog (in member SYSCATLG), and the I/O configuration members built by the MVSCP. It may also contain a pointer to an alternate master catalog in member SYSCATxx.

Requirements for Specification

General Information: SYS1.NUCLEUS is required. This system data set must reside on the system residence volume. Allocate space in cylinders. Do not allocate secondary space. Do not allocate multiple extents. Specify CONTIG in the SPACE parameter. (See Figure 4-5 on page 4-7.)

You may create additional specialized versions of the DAT-on nucleus.

Using the DATASET Statement: You can include a maximum of 10 user-written load modules in SYS1.NUCLEUS during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For space allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.OVERLIB

Contents: SYS1.OVERLIB is a partitioned data set that contains overlays generated by you or by the Overlay Generation Language program.

Requirements for Specification

General Information: This system data set is required only if you install the Overlay Generation Language program with the IBM 3800 Model 3 Printing Subsystem. This system data set must reside on a direct access volume, which can be the system residence volume. When using SYS1.OVERLIB, specify DISP=OLD when running the Overlay Generation Language program, and DISP=SHR at all other times. Specify the volume with

Using JCL and Access Method Services: Allocate space for, and catalog, this data set. Use either access method services or JCL for cataloging.

You may copy the sample JCL (used to create objects) shipped with the Overlay Generation Language program, and residing in the DLIBs, into SYS1.OVERLIB using the existing ACCEPT or APPLY commands in SMP, or with the IEBCOPY utility.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=VBM

LRECL=8205

BLKSIZE=8209

SYS1.PARMLIB

Contents: SYS1.PARMLIB is a partitioned data set that contains IBM-supplied and installation-created lists of system parameter values.

Requirements for Specification

General Information: SYS1.PARMLIB is required. This system data set must reside on a direct access volume, which can be the system residence volume. Do not allocate secondary space.

Using the DATASET statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written system parameters in SYS1.PARMLIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 (Model 2)
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
1/440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Additional Information: For additional information, see *Initialization and Tuning.*

SYS1.PDEFLIB

Contents: SYS1.PDEFLIB is a partitioned data set that contains page definitions built by the Print Management Facility (PMF), or by your installation.

Requirements for Specification

General Information: SYS1.PDEFLIB is required only if you install the Print Services Facility (PSF) with a printer that can be driven by PSF, such as the IBM 3800 Model 3 Printing Subsystem. This system data set must reside on a direct access volume, which can be the system residence volume. When using SYS1.PDEFLIB, specify DISP=SHR.

Using JCL and Access Method Services: Allocate space for and catalog this data set. Use either access method services or JCL for cataloging.

You can copy the page definition objects shipped with PSF, and residing in the DLIBs, into SYS1.PDEFLIB using the existing ACCEPT or APPLY commands in SMP, or with the IEBCOPY utility.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=VBM

LRECL=8205

BLKSIZE=8209

SYS1.PROCLIB

Contents: SYS1.PROCLIB is a partitioned data set that contains the cataloged procedures used to perform certain system functions. The cataloged procedures can be for system tasks or processing program tasks invoked by the operator or the programmer.

Requirements for Specification

General Information: SYS1.PROCLIB is required. This system data set must reside on a direct access volume, which can be the system residence volume. For improved system efficiency, allocate space on a cylinder boundary. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written procedures in SYS1.PROCLIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 which is less than or equal to:

```
14640 for a 2305 (Model 2)
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.PSEGLIB

Contents: SYS1.PSEGLIB is a partitioned data set that contains page segments built by the Print Management Facility (PMF), or by your installation.

Requirements for Specification

General Information: This system data set is required only if you install the Print Services Facility (PSF) with a printer that can be driven by PSF, such as the IBM 3800 Model 3 Printing Subsystem. This system data set must reside on a direct access volume, which can be the system residence volume. When using SYS1.PSEGLIB, specify DISP=SHR.

Using JCL and Access Method Services: Allocate space for, and catalog, this data set. Use either access method services or JCL for cataloging.

Sample source objects are shipped with PSF. You can build these source objects into SYS1.PSEGLIB using PSF.

For allocation, specify the following DCB subparameters on the DD statement:

RECFM=VBM

LRECL=8205

BLKSIZE=8209

SYS1.SAMPLIB

Contents: SYS1.SAMPLIB is a partitioned data set that contains the installation verification procedure (IVP), the independent utilities, and the IPL text. It also contains sample exit routines.

Requirements for Specification

General Information: SYS1.SAMPLIB is required. This system data set must reside on a direct access volume, which can be the system residence volume. You need not catalog this system data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,BLKSIZE=nnnn
```

where nnnn is a multiple of 80, but not greater than 3200.

Additional Information: For information on IVP, see Chapter 6, "Testing the System" on page 6-1.

SYS1.SBLSCLI0

Contents: SYS1.SBLSCLI0 is a partitioned data set that contains the text of the CLISTs for the Interactive Problem Control System (IPCS).

Requirements for Specification

General Information: To use the IPCS dialog programs, install the Interactive System Productivity Facility (ISPF).

This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written IPCS CLISTs in SYS1.SBLSCLI0 during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

Because this data set is a CLIST data set, concatenate it to other CLIST data sets. Specify the same DCB subparameters for this data set as you do for the other CLIST data sets.

If you have not established CLIST data set conventions yet, specify a nominal set of DCB subparameters, such as:

RECFM=VB

LRECL=255

BLKSIZE=3360

SYS1.SBLSKEL0

Contents: SYS1.SBLSKEL0 is a partitioned data set that contains the file tailoring skeletons for the Interactive Problem Control System (IPCS) dialog programs.

Requirements for Specification

General Information: To use the IPCS dialog programs, install the Interactive System Productivity Facility (ISPF).

This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written IPCS dialog file tailoring skeletons in SYS1.SBLSKEL0 during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

Because this data set is an ISPF dialog message data set, concatenate it to other ISPF data sets. Specify the same DCB subparameters for this data set as you do for the other ISPF data sets.

If you have not installed ISPF yet, specify a nominal set of DCB subparameters, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.SBLSMSG0

Contents: SYS1.SBLSMSG0 is a partitioned data set that contains the message text for the Interactive Problem Control System (IPCS) dialog programs.

Requirements for Specification

General Information: To use the IPCS dialog programs, install the Interactive System Productivity Facility (ISPF).

This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written IPCS dialog message members in SYS1.SBLSMSG0 during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

Because this data set is an ISPF dialog message data set, concatenate it to other ISPF data sets. Specify the same DCB subparameters for this data set as you do for the other ISPF data sets.

If you have not installed ISPF yet, specify a nominal set of DCB subparameters, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: Concatenate this data set in front of the ISPF message library.

SYS1.SBLSPNL0

Contents: SYS1.SBLSPNL0 is a partitioned data set that contains the dialog panels for the IPCS dialog programs.

Requirements for Specification

General Information: To use the IPCS dialog programs, install the Interactive System Productivity Facility (ISPF).

This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written IPCS dialog panels in SYS1.SBLSPNL0 during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

Because this data set is an ISPF dialog panel data set, concatenate it to other ISPF data sets. Specify the same DCB subparameters for this data set as you do for the other ISPF data sets.

If you have not installed ISPF yet, specify a nominal set of DCB subparameters, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: Concatenate this data set in front of the ISPF panel library.

SYS1.SBLSTBL0

Contents: SYS1.SBLSTBL0 is a partitioned data set that contains the dialog tables for the IPCS dialog programs.

Requirements for Specification

General Information: To use the IPCS dialog programs, install the Interactive System Productivity Facility (ISPF).

This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written IPCS dialog tables in SYS1.SBLSTBL0 during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

Because this data set is an ISPF dialog panel data set, concatenate it to other ISPF data sets. Specify the same DCB subparameters for this data set as you do for the other ISPF data sets.

If you have not installed ISPF yet, specify a nominal set of DCB subparameters, such as:

RECFM=FB

LRECL=80

BLKSIZE=nnnnn

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.STGINDEX

Contents: SYS1.STGINDEX is a VSAM data set that contains auxiliary storage management records for virtual I/O (VIO) data sets that MVS saves across job steps and between IPLs.

Requirements for Specification

General Information: SYS1.STGINDEX is required only if you want MVS to save VIO data sets across job steps and between IPLs. This system data set must reside on a direct access volume, which can be the system residence volume. Permanently mount or reserve this volume while it is in use. Do not allocate secondary space.

Note: If SYS1.STGINDEX is not in the system during IPL, auxiliary storage management (ASM) issues two messages:

```
ILR001I  rc DYNAMIC ALLOCATION OF SYS1.STGINDEX FAILED,
         NO VIO JOURNALING

ILR022A  REPLY 'U' TO CONTINUE WITHOUT VIO JOURNALING
```

See *System Messages*.

Using the DATASET Statement: Allocate space in cylinders.

Using Access Method Services: If you predefine this system data set, allocate space in cylinders. To predefine SYS1.STGINDEX, specify the following command and parameters with appropriate values where indicated.

```
DEFINE CLUSTER -
 (NAME(SYS1.STGINDEX) -
 FILE(STGINDEX) -
 KEYS(12,8) -
 CYLINDERS(primary) -
 BUFFERSPACE(20480) -
 RECORDSIZE(2041 2041) -
 VOLUME(volser) -
 REUSE) -
DATA (CONTROLINTERVALSIZE(2048)) -
INDEX (CONTROLINTERVALSIZE(4096))
```

Note: Do not share this data set across systems.

Additional Information: The commands and parameters used to predefine SYS1.STGINDEX are discussed in *Access Method Services Reference*.

SYS1.SVCLIB

Contents: SYS1.SVCLIB is a partitioned data set that contains some OLTEP and appendage modules.

Requirements for Specification

General Information: SYS1.SVCLIB is required. This system data set must reside on the system residence volume. Allocate space in cylinders; you may allocate secondary space.

Using the DATASET Statement: If you do not specify this system data set on a DATASET statement, it will reside on the volume you specify in the RESVOL parameter on the GENERATE statement (the system residence volume).

You can include user-written routines, in load module form, during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.TCOMMAC

Contents: SYS1.TCOMMAC is a partitioned data set that contains ACF/TCAM record API macros.

Requirements for Specification

General Information: SYS1.TCOMMAC is required only if you install ACF/TCAM. This data set must reside on a direct access volume, which can be the system residence volume. You need not catalog this data set. You may allocate secondary space.

Using the DATASET Statement: Specify this data set on a DATASET statement only if you install ACF/TCAM.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=FB,LRECL=80,
BLKSIZE=nnnnn
```

where nnnnn is a multiple of 80 that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.TELCMLIB

Contents: SYS1.TELCMLIB is a partitioned data set that contains telecommunications subroutines in load module form.

Requirements for Specification

General Information: SYS1.TELCMLIB is required only if you install BTAM/SP and/or ACF/TCAM. This system data set must reside on a direct access volume, which can be the system residence volume. For improved system efficiency, allocate space on a cylinder boundary. You may allocate secondary space.

Using the DATASET Statement: If you install BTAM/SP and/or ACF/TCAM, specify this data set on a DATASET statement, even if you predefine it. If you predefine it, omit the SPACE parameter.

You can include user-written routines, in load module form, during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,
BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

SYS1.UADS

Contents: SYS1.UADS is a partitioned data set that contains a list of authorized TSO users, and information about them.

Requirements for Specification

General Information: SYS1.UADS must reside on a direct access volume, which can be the system residence volume. Catalog this system data set, but allocate space only if you activate TSO. You may allocate secondary space.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

TSO	TSO/E
DSORG=PO,LRECL=80,RECFM=FB, BLKSIZE=nnnnn	DSORG=PO,LRECL=172,RECFM=FB, BLKSIZE=nnnnn

where nnnnn is a multiple of LRECL that is less than or equal to:

```
14640 for a 2305 Model 2
12960 for a 3330 Model 1, 2, or 11
12960 for a 3333 Model 1 or 11
 8320 for a 3340/3344
19040 for a 3350
17440 for a 3375
23440 for a 3380
27920 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Notes:

1. Do not specify an expiration date for SYS1.UADS.
2. Do not share this data set across systems unless GRS is installed and active.

For a detailed explanation of resource sharing in this regard, see *TSO/E Customization*.

SYS1.VTAMLIB

Contents: SYS1.VTAMLIB is a partitioned data set that contains the ACF/VTAM load modules, installation-coded logon exit routines, authorization and accounting exit routines, USS definition tables, and the network solicitor.

Requirements for Specification

General Information: SYS1.VTAMLIB is required only if you install ACF/VTAM. This system data set must reside on a direct access volume, which can be the system residence volume. Catalog this data set. You may allocate secondary space.

Using the DATASET Statement: If you install ACF/VTAM, specify this data set on a DATASET statement.

You can include user-written routines, in load module form, in SYS1.VTAMLIB during sysgen.

Using JCL and Access Method Services: Use either for cataloging.

For allocation, specify the following DCB subparameters on the DD statement:

```
RECFM=U,BLKSIZE=nnnnn
```

where nnnnn is:

```
14660 for a 2305 Model 2
13030 for a 3330 Model 1, 2, or 11
13030 for a 3333 Model 1 or 11
 8368 for a 3340/3344
18432 for a 3350
17600 for a 3375
18432 for a 3380
18432 for a 3390
```

The preceding values represent optimal space utilization for the above devices.

Note: SYS1.VTAMLIB, as well as the user-defined network control program load module library, must be authorized. Therefore, include both libraries in the IEAAPFxx PARMLIB member.

Page and Swap Data Sets

Defining the Page and Swap Data Sets

This section contains general information about defining the page and swap data sets using either the DATASET macro or the access method services DEFINE command.

Page or swap data sets contain the paged-out portions of address spaces, the common service area (CSA), and the data written to virtual I/O data sets.

Defining a page or swap data set consists of allocating space for it on a direct access volume, cataloging it, formatting it, and including its name in the PAGE= or SWAP= portion of either the IEASYS00 or the IEASYSnn system parameter member in SYS1.PARMLIB.

At least 3 page data sets must be defined during sysgen. No swap data sets are required. Up to 25 page data sets and up to 10 swap data sets may be defined before or during sysgen, and additional page or swap data sets may be defined after sysgen. The page or swap data sets that are defined during sysgen are automatically included in the IEASYS00 member in SYS1.PARMLIB. For those page or swap data sets defined before sysgen, specify the name of the page or swap data set in a DATASET macro and then, during sysgen, the name will be included in the IEASYS00 member. For those page or swap data sets defined after sysgen, enter their names in either the IEASYS00 or IEASYSnn member. (Do this either by using the system utility IEBUPDTE before IPL, or by entering the name from the console in the PAGE=keyword during IPL.)

If page data sets from a previously generated system are to be reused, it is necessary to reuse the master catalog as well. See Chapter 7, "Using Data Sets from the Old Production System" on page 7-1.

The first-named page data set in either the IEASYS00 member or the IEASYSnn member in SYS1.PARMLIB will be used for the pageable link pack area (PLPA). For IEASYS00, this means that the data set name appearing in the first DATASET macro that contains the PAGEDSN parameter will be used for the PLPA. The second-named page data set in either the IEASYS00 member or the IEASYSnn member in SYS1.PARMLIB will be used as the system data set for the pages for the common areas.

For information about adding page or swap data set names to SYS1.PARMLIB and about system parameter members, see *Initialization and Tuning*.

Defining Page or Swap Data Sets, Using the DATASET Macro

The DATASET macro can be used to define the page or swap data sets during sysgen. One macro must be specified for each page or swap data set to be defined. At least 3 page data sets must be defined. The parameters for the DATASET macro are discussed in Chapter 3, "Coding the Sysgen Statements" on page 3-1. When the DATASET macro is used to define a page or swap data set, its name is automatically included in the PAGE=, SWAP=, or DUPLEX= portion of the IEASYS00 member in SYS1.PARMLIB.

Even if predefining the page or swap data sets, use the DATASET macro to include the name of the page data sets in IEASYS00 as follows:

```
Name  DATASET  PAGEDSN=dsname
Name  DATASET  SWAPDSN=dsname
Name  DATASET  DUPLEXDS,NAME=dsname
```

Absence of the SPACE parameter prevents space allocation but allows the page or swap data set name to be entered in the IEASYS00 member in SYS1.PARMLIB. When the SPACE parameter is not specified, the VOL parameter need not be specified. If VOL is specified without the SPACE parameter, VOL will be ignored.

When using the DATASET macro to define page and swap data sets, space can only be allocated in cylinders and only primary extents can be specified.

Figure 4-1 on page 4-2 gives an example of the use of the DATASET macro to define the page and swap data sets.

Defining Page or Swap Data Sets Using Access Method Services

The page and swap data sets can be defined before sysgen using the DEFINE command of access method services.

In allocating space using access method services, it is recommended that your allocation be in cylinders. Only primary extents may be specified for page or swap data sets.

Figure 4-6 on page 4-11 shows the input for using access method services for defining page and swap data sets as well as system data sets.

The DEFINE command does not include the data set names in the PAGE=, SWAP=, or DUPLEX= portion of the IEASYS00 member in SYS1.PARMLIB.

Protecting Page and Swap Data Sets

Page and swap data sets should be password protected to prevent unauthorized use. During sysgen, the page and swap data sets are defined without password protection. At the successful conclusion of the sysgen process, update the master catalog to protect the page space data sets. The output from job 6 of a complete sysgen includes a listing of the master catalog. The names and data components of the page or swap space can be determined from this listing. Alternatively, the access method services LISTCAT command can be issued to determine the page or swap space and data component names of the data sets. The ALTER commands that follow will alter the catalog entries for a page data set. *Access Method Services Reference* provides additional details on the format and function of the LISTCAT and ALTER commands.

```
//ALTPSWD     JOB  ACCT123,PROGRAMMER,MSGLEVEL=(1,1)
//STEP1       EXEC PGM=IDCAMS
//SYSPRINT    DD   SYSOUT=A
//SYSIN       DD   *
    ALTER -
         SYS1.pagedsn -
         MASTERPW(password) -
         CONTROLPW(password) -
         UPDATEPW(password) -
         READPW(password) -
         AUTHORIZATION(entrypoint[string]) -
         ATTEMPTS(0)
    ALTER -
         frstqual.Tbbbbbbb.DFDyyddd.Taaaaaaa.Tbbbbbbb -
         MASTERPW(password) -
         CONTROLPW(password) -
         UPDATEPW(password) -
         READPW(password) -
         AUTHORIZATION(entrypoint[string]) -
         ATTEMPTS(0)
/*
```

Note: *pagedsn* is the name of a page data set as cataloged during sysgen.

The name of the data component is generated by the system and is listed in the LISTCAT output. It is always in the form

frstqual.Tbbbbbbb.DFDyyddd.Taaaaaaa.Tbbbbbbb,

where:

frstqual
is the first qualifier of the cluster component.

yy is the year.

ddd
is the day.

aaaaaaa and bbbbbbb
are the time of definition of the page or swap data set.

For data sets cataloged in VSAM catalogs, **frstqual** is always VSAMDSET.

Chapter 5. Executing Sysgen

This chapter contains four sections:

- "Stage 1: Producing the Job Stream" explains Stage 1 input, processing, and output: the production of the job stream for Stage 2
- "Stage 2: Processing the Job Stream" on page 5-4 explains Stage 2 input, processing of the job stream and output: the production of a generated system
- "Restart Procedures" on page 5-10 explains the possible causes for unsatisfactory completion of sysgen and suggested restart procedures for both Stage 1 and Stage 2
- "Sysgen Messages" on page 5-13 gives examples of sysgen error, warning, and informational messages that you might encounter

Stage 1: Producing the Job Stream

Stage 1 Input

The input statements required for Stage 1 consist of JCL and sysgen statements. (See Figure 5-1 on page 5-3 for a sample of Stage 1 input.)

- A JOB statement with all required parameters.
- An EXEC statement with PGM=IEV90,REGION=2048K. IEV90 is the name assigned to Assembler H Version 2. In the example, 2048K is a sample REGION size.
- A SYSLIB DD statement that allocates the SYS1.AGENLIB and SYS1.AMODGEN libraries to this assembly.
- A SYSUT1 DD statement that allocates space for the temporary data set required by the assembler for Stage 1.
- A SYSPUNCH DD statement that defines the data set that is to contain the job stream produced during Stage 1.
- A SYSPRINT DD statement that defines the output data set or class.
- A SYSIN DD * statement.
- The sysgen statements.
- An END statement.
- A /* statement.

The sequence of the statements is shown in Figure 5-1.

```
//SYSGEN    JOB MSGLEVEL=1
//START     EXEC PGM=IEV90,REGION=2048K
//SYSLIB    DD DSNAME=SYS1.AGENLIB,DISP=SHR
//          DD DSNAME=SYS1.AMODGEN,DISP=SHR
//SYSUT1    DD UNIT=SYSDA,SPACE=(cyl,(5,5))
//SYSPUNCH  DD UNIT=unitadr,LABEL=(,NL)
//SYSPRINT  DD SYSOUT=A
//SYSIN     DD *
       sysgen statements
           END
/*
```

Figure 5-1. Sample Stage 1 Input

Stage 1 Processing

The assembler processes the sysgen statements as one job during Stage 1. The statements are analyzed for valid parameters and dependencies upon other statements.

If the assembler finds no errors, it generates a job stream consisting of JCL and control statements for Stage 2, places /* in columns 1 and 2 of the last record, and sets a return code of 0.

If the assembler finds errors, it writes error messages with severity codes of 5 or 7 (see "Sysgen Messages" on page 5-12) and normally does not generate the job stream.

Diagnostic Override

Diagnostic override enables the assembler to generate a job stream during Stage 1 even though it finds errors in the Stage 1 input statements. The assembler identifies the errors by placing // in columns 1 and 2 of the last record. Any job stream the assembler generates is invalid.

To override errors, include the following statements in the Stage 1 input statements after the Stage 1 JCL statements:

```
            COPY  SGSYSPAK
&SGMENTB(16) SETB  1
```

Stage 1 Output

The assembler generates the job stream as punched cards, card images on tape, or card images on a DASD volume, according to your specification on the Stage 1 SYSPUNCH DD statement. In addition to the job stream, the assembler generates a documentation listing, which is a printout of the processing of all the statements, including the PUNCH statements that comprise the input to Stage 2.

Stage 2: Processing the Job Stream

When Stage 1 completes and the assembler generates a valid job stream, Stage 2 can begin.

Stage 2 Input

The input to Stage 2 is the job stream output from Stage 1.

The Job Stream

The job stream contains several JOB statements, each followed by one or more EXEC statements. Each EXEC statement is followed by its associated DD statements and other data required to execute the linkage editor, utility programs, and access method services.

The JOB Statement: The format of the Stage 2 JOB statement is:

```
//name_n  JOB 1,'SYSTEM GENERATION',MSGLEVEL=1,
//          CLASS=&SGCTRLC(42),MSGCLASS=&SGCTRLC(41)
```

where "name" is the value from the INDEX parameter on the GENERATE statement, and "n" represents the sequential identification numbers supplied Stage 2 to indicate the number of the job (0 through 6).

The value you specify for the JCLASS parameter on the GENERATE statement is substituted for the global &SGCTRLC(42) for the job class. The value you specify for the OCLASS parameter on the GENERATE statement is substituted for the global &SGCTRLC(41) for the output class for system messages.

Note: For job 6 of Stage 2, in addition to the parameters described above, the JOB statement includes the TYPRUN=HOLD parameter to enable MVS to hold the job indefinitely. (See "Initializing the New Master Catalog" on page 5-7 and "Stage 2 Processing of the Job Stream" on page 5-5.)

You may modify JOB statements by changing the jobname or the accounting information on the JOB statement before Stage 1. Use the IEBUPDTE utility program or an editor to modify the JOB statement information contained in member JOBCARD in SYS1.AGENLIB distribution library. (For information on using the IEBUPDTE utility program, see *Utilities*.)

When modifying the JOB statement, specify the NAME parameter as:

```
jobname&SGCTRLA(3)
```

where:

jobname consists of 1 through 6 characters, either alphameric, #, @, or $ with the first character being either alphabetic, #, @, or $;

and

&SGCTRLA(3) is a counter, incremented by the assembler, that produces unique jobnames.

(For information on coding a JOB statement, see *JCL Reference*.)

Figure 5-2 is an example of creating unique JOB statements. The TYPRUN=HOLD parameter is present on the last job statement, as required when the system uses more than one initiator to process several jobs.

```
Col

1 3     9

//JOBA     JOB  (78700,J22),NAME,MSGLEVEL=(1,1)
//UPDTE    EXEC PGM=IEBUPDTE,PARM=MOD
//SYSPRINT DD   SYSOUT=A
//SYSUT1   DD   DSN=SYS1.AGENLIB,DISP=OLD
//SYSUT2   DD   DSN=SYS1.AGENLIB,DISP=OLD
//SYSIN    DD   *
./  REPL   NAME=JOBCARD,LIST=ALL
./  NUMBER NEW1=10000,INCR=10000
         MACRO
         JOBCARD
         COPY  SGGBLPAK
 PUNCH '//SYS#10&SGCTRLA(3) JOB (USER ACCG.),''USER DESCRIPTION'','
         AIF   (&SGHOLDB EQ 1).HOLD   IF JOB IS TO BE HELD
 PUNCH '//     MSGLEVEL=1,MSGCLASS=C,CLASS=C'
         AGO   .MEND
.*****************                                    ******************
.****       PUNCH 'HOLD' FOR JOBS WHICH MUST BE HELD                 ****
.*****************                                    ******************
.HOLD    ANOP
 PUNCH '//     MSGLEVEL=1,MSGCLASS=C,CLASS=C,TYPRUN=HOLD'
.MEND    ANOP
&SGCTRLA(3) SETA &SGCTRLA(3)+1            INCREMENT JOB NUMBER
&SGCTRLA(1) SETA 0                        INITIALIZE STEP NUMBER
         MEND
./
/*
//
```

Figure 5-2. Example of a Macro to Create JOB Statements for Stage 2 Input

The EXEC Statement: The format of the Stage 2 EXEC statements produced by Stage 1 is:

```
//name EXEC PGM=program[,parameter(s)]
```

or

```
//name EXEC procname[,parameter(s)]
```

The step name is either

- SG*nn* (for steps produced by internal macros), or
- *abbnn* (for steps produced by the GENERATE statement), where *abb* is an abbreviation of the name of the target system data set (for example, LNK for SYS1.LINKLIB). In job 5, where Stage 2 creates many system data sets, the step name is MISC.

The suffix *nn* in both formats indicates the sequential number of the step within the job. The number of steps in a job may vary according to what you specified on the statements in Stage 1.

Service and utility programs executed during Stage 2 include: IDCAMS, IEBCOPY, IEBGENER, IEBUPDTE, IEHLIST, IFCDIP00. (IDCAMS is described in *Access Method Services Reference*. IFCDIP00 is described in *SYS1.LOGREC Error Recording*. The remaining programs are described in *Utilities*.)

The name of the supplied procedure executed is LINKS for the linkage editor.

Stage 2 Processing

Execution of Stage 2 requires a job stream of seven jobs (or six jobs if you previously defined the master catalog).

During execution of the job stream, Stage 2 creates each system data set and page data set (if specified) within a single job. During the process of creating a data set, the following may occur:

- Definition of the system or page data set (if you included a DATASET statement).
- Link-editing of selected modules to construct members of the system data sets.
- Completing the construction of the system data sets using utility programs and service aids.

Stage 2 Processing of the Job Stream

The job stream consists of seven jobs (or six jobs, if you previously defined the master catalog). If you do not use the DATASET statement to define the system and/or page data sets, Stage 2 does not allocate space for them.

Job 0: During job 0, define the master catalog. (Only include this job in the input stream if you have not previously defined the master catalog.).

Job 1: During job 1, create SYS1.LPALIB and, optionally, SYS1.ISAMLPA as follows:

- Define SYS1.LPALIB and SYS1.ISAMLPA.
- Copy components from the DLIBs and from user-defined data sets into SYS1.LPALIB and SYS1.ISAMLPA.
- Link-edit components from the DLIBs into SYS1.LPALIB.

Job 2: During job 2, create SYS1.LINKLIB as follows:

- Define SYS1.LINKLIB.
- Copy components from the DLIBs and from user-defined data sets into SYS1.LINKLIB.
- Link-edit components from the DLIBs into SYS1.LINKLIB.

Job 3: During job 3, create SYS1.SVCLIB as follows:

- Define SYS1.SVCLIB.
- Copy components from the DLIBs and from user-defined data sets into SYS1.SVCLIB.

Job 4: During job 4, create SYS1.NUCLEUS as follows:

- Define SYS1.NUCLEUS.
- Copy components from the DLIBs and from user-defined data sets into SYS1.NUCLEUS.
- Link-edit components from the DLIBs and user-defined data sets into SYS1.NUCLEUS.

Job 5: During job 5, create the remaining system data sets and the page data sets as follows:

- Define the remaining non-VSAM system data sets (including the TSO system data sets, if specified).
- Execute the IFCDIP00 program to initialize SYS1.LOGREC.
- Copy components from the DLIBs and from user-defined data sets into specific system data sets.
- Execute the IEBUPDTE utility program to add members to SYS1.PARMLIB, according to the specifications you made on the DATASET statement.
- Define the page data sets and the SYS1.STGINDEX VSAM system data set.

Job 6: Do not execute this job until jobs 0 through 5 run successfully. During job 6, the catalog Stage 2 becomes the master catalog of the new system, and the following occurs:

- Products that you have previously installed and that have sysgen support (for example, RACF and DFHSM) are copied and link-edited into the system libraries from the DLIBs.
- The VTOCs of all volumes containing the new system data sets are listed.

- The directories of the system data sets are listed.
- The master catalog is listed.
- The old master catalog pointer (in SYS1.NUCLEUS) for the generating system is deleted (scratched).
- The pointer to the new master catalog is initialized. The volume serial number of the new master catalog is placed into member SYSCATLG of SYS1.NUCLEUS, thereby making the new catalog the master catalog.

After Stage 2 you can change the master catalog pointer in SYS1.NUCLEUS by re-executing the steps in job 6. (For additional information about job 6, see "Initializing the New Master Catalog" on page 5-7.)

Job 0 executes first. The first step of job 0 defines the master catalog. After creating the master catalog, jobs 1, 2, and 3 can execute simultaneously. Because these jobs are different sizes, they may finish out of sequence.

Note: Job 6 does not begin execution until the operator releases it from the queue (see "Initializing the New Master Catalog" on page 5-7).

Initializing the New Master Catalog

Job 6 of Stage 2 initializes the master catalog. The JOB statement for job 6 includes the TYPRUN=HOLD parameter, which keeps job 6 from executing until released by the operator. The operator releases job 6 by issuing the command: $A 'SYSGEN6'

IMPORTANT: Do not release job 6 until jobs 0 through 5 complete successfully. Job 6 initializes the pointer to the new master catalog and disconnects the generating system's master catalog by:

- Deleting the pointer in the generating system's master catalog that points to the new system's master catalog and
- Storing a pointer in the new SYS1.NUCLEUS that points to the new system's master catalog.

Some installations will want to use the master catalog from a previous production system. Chapter 7, "Using Data Sets from the Old Production System" on page 7-1 contains an example and a procedure for combining the system data sets from a newly generated system with the master catalog, SYS1.STGINDEX, and page data sets from an existing production system.

Multiprogramming the Job Stream

The design of Stage 2 allows you to multiprogram the jobs in the job stream.

After MVS reads in and enqueues the job stream, start from one to three initiators to begin execution. (Start no more than three initiators for a Stage 2.) The following text assumes three initiators; it also assumes that you define the master catalog using a DATASET statement.

Job 0 executes first. The first step of Job 0 defines the master catalog. After creating the master catalog, the remainder of jobs 1 through 5 execute simultaneously, as initiators are freed. Because these jobs are different sizes, they may finish out of sequence. Job 6 does not begin execution until released from the queue by the operator (see "Initializing the New Master Catalog" on page 5-7).

Stage 2 Output

The output from Stage 2 is the new and updated system data sets.

Stage 2 also produces a documentation listing that is a printout of all the steps executed by the linkage editor, the utility programs, and access method services.

Restart Procedures

This section contains the information necessary to restart a sysgen after termination of the process by errors in either Stage 1 or 2.

Restarting Stage 1

The most common causes of error during Stage 1 are data entry errors in the input statements, and contradictory or invalid specifications on the statements.

The assembler generates error messages to indicate the type of error and writes them to the SYSPRINT data set (see "Sysgen Messages" on page 5-12).

If the assembler finds any errors during Stage 1, it does not generate the job stream. (For exceptions, see "Diagnostic Override" on page 5-2.)

In Stage 1, the assembler processes the sysgen statements for the first time. Therefore, you can only restart Stage 1 from the beginning. To restart, correct the errors in the input statements, and resubmit the job.

Restarting Stage 2

The most common causes of error during Stage 2 are:

- Faulty space allocation for the system data sets
- Errors in the input statements that the assembler cannot detect during Stage 1
- Procedural errors, such as improper volume mounting

The most probable error during Stage 2 is faulty space allocation for a system data set. If this occurs, reallocate space for that system data set, increase its size, and re-execute that job starting at the beginning.

You can use Device Support Facilities to reinitialize the volume that is to contain the new master catalog.

If you restart with job 0, which defines a master catalog, you can use the access method services EXPORT command, as shown in the following example, to delete the new master catalog's name from the generating system's catalog.

```
//DELCATNM    JOB ACCT123,PROGRAMMER,MSGLEVEL=(1,1)
//STEP1       EXEC PGM=IDCAMS
//SYSPRINT    DD  SYSOUT=A
//SYSIN       DD  *
    EXPORT -
         SYS1.A.NEW.ICF.CATALOG -
         DISCONNECT
/*
```

If the system data set contains data, you can save the data and restart Stage 2 from the failing job step by doing the following:

1. Rename the system data set, using IEHPROGM.
2. Reallocate space for the system data set using its original name.
3. Copy the data in the renamed data set into the reallocated system data set using IEBCOPY.
4. Delete the renamed data set using IEHPROGM.
5. Restart Stage 2 from the beginning of the job step that failed. You can accomplish the actual restarting by using one of the methods described in "Restart Techniques."

If the problem you encounter is one other than space allocation, such as component failure or machine malfunction, follow the instructions printed out in the error messages and codes and then see *System Messages* or *System Codes*.

For information on IEHPROGM and IEBCOPY, see *Utilities*. For information on allocating space for system data sets, see Chapter 4, "Selecting and Defining the System Data Sets" on page 4-1 and *Access Method Services Reference*.

Restart Techniques

You can restart Stage 2 at the beginning of any job or job step. If you do not need to change any statements, restart the job stream from the job or job step that failed. This section discusses the techniques used for restarting the job stream after you perform any other necessary operations.

Restarting from Cards

If the job stream is on cards, you can restart a job step by placing a JOB card ahead of the job step's EXEC card and beginning from that job step.

Punching the Job Stream

If the unit (SYSPUNCH) you specified during Stage 1 for the job stream was not a card punch, and you want the job stream on cards, use the IEBEDIT utility to punch all or some of the job stream. Figure 5-3 on page 5-11 shows the statements required to punch the job stream using IEBEDIT.

When you use the IEBEDIT utility program to punch any part or all of the job stream, consider the following:

- The value of the UNIT parameter of the SYSUT1 DD statement is the unit name (generic or esoteric) of the device on which the job stream resides. If the job stream is on disk, specify a standard label in the LABEL parameter.
- You can specify one or more EDIT statements when executing IEBEDIT. You can also elect to punch certain jobs or job steps. For the control statements for punching certain jobs or job steps, see Figure 5-4 on page 5-11.

```
//PUNCH    JOB ACCOUNT123
//         EXEC PGM=IEBEDIT
//SYSPRINT DD SYSOUT=A
//SYSUT1   DD UNIT=unitname,LABEL=(,NL),VOLUME=SER=volser,
//           DISP=(OLD,KEEP),DCB=(dcb information)
//SYSUT2   DD SYSOUT=B
//SYSIN    DD *
         EDIT START=name1,STEPNAME=xxx
         EDIT START=name2,STEPNAME=xxx
         EDIT START=name3,STEPNAME=xxx
         EDIT START=name4,STEPNAME=xxx
         EDIT START=name5,STEPNAME=xxx
         EDIT START=name6,STEPNAME=xxx
/*
```

Figure 5-3. Punching the Job Stream

Restarting from Tape or a DASD Volume

You can also use the IEBEDIT utility to restart Stage 2 from any job or job step when the job stream is on tape or a DASD volume.

IEBEDIT creates a new job stream by editing and selectively copying the job stream provided as input. IEBEDIT can copy an entire set of jobs, including JOB statements and associated job step statements, or selected job steps in a job. Figure 5-4 shows the control statements required by IEBEDIT.

```
//RESTART  JOB
//         EXEC PGM=IEBEDIT
//SYSPRINT DD SYSOUT=A
//SYSUT1   DD UNIT=3400-6,LABEL=(,NL),
//            VOLUME=SER=volser,DISP=(OLD,KEEP),
//            DSN=dsname,
//            DCB=(DCB information)
//SYSUT2   DD SYSOUT=(A,INTRDR)
//SYSIN    DD *
   EDIT    START=namen,STEPNAME=SGxx[,NOPRINT]
/*
```

Figure 5-4. Control Statements for IEBEDIT when the Job Stream is on Tape

When you use the IEBEDIT utility program to create a job stream to permit the restart of Stage 2, consider the following:

- The value of the UNIT parameter of the SYSUT1 DD statement is the unit address of the magnetic tape drive or direct access storage device on which the job stream resides.
- SYSUT2 is directed to an internal reader. The new job stream goes directly into the job queue for execution.
- You can specify one or more EDIT statements when executing IEBEDIT. If you omit the TYPE parameter, as in Figure 5-4, STEPNAME specifies the first job step in the job specified by the START parameter included in the new job stream. If you specify TYPE=INCLUDE or TYPE=EXCLUDE, STEPNAME specifies the job steps you want included or excluded, respectively, from the new job stream. You can specify individual job steps and sequences of job steps for inclusion or exclusion.
- Include NOPRINT, if you do not want a listing of the new job stream.

- If the SYSUT2 DD statement designates a magnetic tape drive or direct access storage device, issue a START RDR command after the creation of the new job stream.

For additional information on the IEBEDIT utility program, see *Utilities*.

Sysgen Messages

The assembler produces messages during the processing of sysgen statements and prints them in the assembler listing in the SYSPRINT data set. The assembler produces three types of messages: error messages, warning messages, and informational messages.

Figure 5-5 shows the severity code and format for error and warning messages.

```
s,***IEIaaannn text


     s = Severity code:
         *     Informational message; documents options
               selected for new system.
         ***   Informational message; appears before any PUNCH
               statements produced by statement processing.
         0     Warning message; the condition
               indicated may cause errors in the new
               system.
         5     Error message; user error in coding of
               a sysgen statement.
         7     Error message; message is produced by
               the GENERATE statement.
   aaa = An abbreviation of the sysgen
         statement on which the error
         was detected:

         aaa          Statement

         DAT          DATAMGT
         DTS          DATASET
         GEN          GENERATE
         JES          JES

   nnn = Message serial number
  text = Message text
```

Figure 5-5. Format of Sysgen Error and Warning Messages

Error Messages

The text for error messages indicates a coding error on a statement.

Examples of error messages are:

Example 1:

```
7,***IEIGEN113 QUIT SWITCH SET PRIOR TO GENERATE MACRO
```

Explanation: One or more errors, indicated by messages, were detected before processing the GENERATE statement.

Severity: 7

User Response: Correct the error or errors indicated, and begin the sysgen process from the start of Stage 1.

Example 2:

```
***IEIGEN116 QUIT SWITCH SET IN GENERATE MACRO
```

Explanation: One or more errors were detected during the processing of the GENERATE statement.

Severity: 7

User Response: Correct the error or errors indicated, and begin the sysgen process from the start of Stage 1.

Example 3:

```
7,***GENERATION TERMINATED***
```

Explanation: The sysgen process was abnormally terminated.

Severity: 7

User Response: None. This message follows message IEIGEN113 and/or IEIGEN116.

Warning Messages

The text for warning messages indicates a condition on a statement that may cause errors in the new system.

Following is an example of a warning message:

Example:

```
0,***IEIDTSW01 VOLUME OR UNIT SUBPARM OF VOL PARM
     IS NULL, WILL BE DEFAULTED TO RESVOL
```

Explanation: DATASET macro for a system data set did not specify a volume or unit. Sysgen assumes the data set is on the system residence volume.

Severity: 0

User Response: None, unless the default is unacceptable.

Informational Messages

Two types of informational messages may occur. Their formats are as follows:

1. ***, message text**
2. *****macro name***component name***component ID*****

The first type of message documents the options you selected for the new system through the sysgen statements. All options are described, whether your selection was explicit or implicit.

The second type of message appears before any PUNCH statements that the assembler generated processing a sysgen statement. The macro name is the name of the sysgen statement that produced any succeeding PUNCH statements. The component name is 3 characters representing the name by which the component is commonly referred; for example, ICF is the component name for the ICF catalog. The component ID is the identification you use to report a problem if you isolate the problem to the code produced by the particular statement.

Chapter 6. Testing the System

The installation verification procedure (IVP) is a program that tests whether the newly installed system is operational. Test any other programs for which verification is required by using their own testing procedures or sample programs.

IVP Minimum Configuration

The IVP job stream executes with all supported machine configurations and requires only the minimum machine configuration.

The IVP Job Streams

The IVP job streams are contained in your system in the partitioned data set SYS1.SAMPLIB. The member names are IVPJOBS, for testing MVS/SP, and DFPX1IVP, DFPX2IVP, and DFPX3IVP, for testing MVS/DFP. The IVP job streams consist of jobs that:

- Execute an assemble/link-edit/go to test the basic functions of the newly installed system
- Provide a listing of selected SYS1.PARMLIB members
- Force an 806 ABEND (see *System Codes*)
- Define catalogs and test basic VSAM functions

Procedures for Using IVP

Once you have carried out the steps outlined in Chapter 2 and built a new system, use IVP to verify that the the system is operational.

Running the MVS/SP IVP Job

1. IPL the system.
2. Issue a START RDR command to the MVS/SP IVP job stream (IVPJOBS). You can use the following command:

   ```
   START RDR.R,132,VOL=SER=SYSRES,LABEL=(,SL),
   DSNAME=SYS1.SAMPLIB(IVPJOBS)
   ```

   If 132 and SYSRES are not the unit address and serial number of the volume on which SYS1.SAMPLIB resides, enter the correct information.
3. The JOB card on IVPJOBS is

   ```
   //IVPJOB JOB PS40230301,MSGLEVEL=1,MSGCLASS=A,REGION=256K
   ```

   If this JOB card does not meet the requirements for your installation, modify the JOB card in SYS1.SAMPLIB(IVPJOBS) before issuing the START RDR command.
4. Review the results of the assemble/link-edit/go for errors.
5. Verify that an 806 ABEND occurred.
6. Save the IVP output for future reference.

Running the MVS/DFP IVP Jobs

1. IPL the system
2. Log onto TSO
3. Edit SYS1.SAMPLIB(DFPX1IVP). You will need to change the volsers of the volumes used to volumes you have available. Also change the catalog name to point to a valid catalog.
4. DFPX1IVP contains three jobs. The second two are held and must be released to run in order.
5. Submit the job, DFPX1IVP. (If the JOB card does not meet the requirements for your installation, modify it before submitting the job.)
6. Edit SYS1.SAMPLIB(DFPX2IVP) and SYS1.SAMPLIB(DFPX3IVP). Again change the volsers and the catalog name, as you did for DFPX1IVP.
7. Submit the jobs DFPX2IVP and DFPX3IVP.
8. Review the output of all three IVPs.
9. Save the output for future reference.

The completion of the above steps and normal end-of-job (EOJ) of the IVP job streams (excluding the 806 ABEND) constitutes successful verification of your new system.

Testing ISMF

This section shows a sequence of panels you can use to verify that ISMF has been installed correctly.

Before you test ISMF, be sure you have completed the steps described in Chapter 8, "Installing ISMF" on page 8-1, for making ISMF available to TSO/E users.

The steps in verifying installation are:

1. Log on to TSO.
2. Display the ISPF/PDF Primary Option Menu.

```
                        ISPF/PDF PRIMARY OPTION MENU
OPTION  ===>

 0   ISPF PARMS  - Specify terminal and user parameters    USERID   - L841557
 1   BROWSE      - Display source data or output listings  TIME     - 14:50
     .
     .
     .
 C   CHANGES     - Display summary of changes for this release
 I   ISMF        - Invoke interactive storage management facility
 T   TUTORIAL    - Display information about ISPF/PDF
 X   EXIT        - Terminate ISPF using log and list defaults
```

3. From the ISPF/PDF Primary Option Menu, enter the option I. This displays the ISMF Primary Option Menu.

   Note: If you cannot invoke ISMF from the ISPF/PDF Primary Option Menu, you do not have ISMF installed correctly. For more information, refer to "Reasons Why ISMF Could Have Failed" on page 6-4.

```
                              ISMF PRIMARY OPTION MENU
COMMAND ===>

SELECT ONE OF THE FOLLOWING:

 0  ISMF PROFILE                 - Change ISMF user profile
 1  DATA SET                     - Perform Data Set Application
 2  VOLUME                       - Perform Volume Application
```

4. From the ISMF Primary Option Menu, enter **PANELID** on the command line. This redisplays the ISMF Primary Option Menu with the panel identification number DGTSMMD1 shown in the left-hand corner:

```
DGTSMMD1                    ISMF PRIMARY OPTION MENU
COMMAND ===>

SELECT ONE OF THE FOLLOWING:

 0  ISMF PROFILE                 - Change ISMF user profile
 1  DATA SET                     - Perform Data Set Application
 2  VOLUME                       - Perform Volume Application
```

5. From the DGTSMMD1 panel, enter **HELP** on the command line. This causes the ISMF Overview Panel (DGTHMD01) to be shown:

```
DGTHMD01                            ISMF OVERVIEW                          HELP
COMMAND ===>

   The Interactive Storage Management Facility (ISMF) of MVS/XA Data Facility
   Product is an ISPF application that helps you manage data set storage
```

Reasons Why ISMF Could Have Failed

If the verification of ISMF installation failed, make sure you executed the new CLIST or used the new LOGON JCL before attempting the verification process.

You can recheck your installation process by retracing the installation steps until you find the cause of failure:

1. Log on to TSO.
2. Execute the CLIST or the LOGON JCL.
3. Make sure that the SMP APPLY processing was completed without errors.
4. Make sure that the SMP RECEIVE processing was completed without errors.
5. Make sure you have followed the steps described in Chapter 8, "Installing ISMF" on page 8-1 for making ISMF available to TSO/E users.

If you have verified all the above steps without finding errors, contact IBM for further assistance.

Chapter 7. Using Data Sets from the Old Production System

You may want to substitute some of the data sets from your old production system for those in the newly generated system. This substitution can make it unnecessary for you to recatalog the data sets and change procedures that you have in the old system, but are not cataloged in the new system.

The example that follows describes a procedure in six steps for reconfiguring your system, substituting the master catalog, page data sets, and SYS1.STGINDEX data set from the old production system for their equivalent members in the newly generated system. Before using a procedure such as this, execute the installation verification procedure (IVP) to ensure that the new system is operational (see Chapter 6, "Testing the System" on page 6-1). Figure 7-1 on page 7-4 shows selected data sets from an old production system on four volumes being combined with selected data sets from a newly generated system on three volumes.

The example that follows assumes that none of the volumes on the old production system or on the newly generated system contain both VSAM and non-VSAM data sets. If VSAM and non-VSAM data sets reside on any volume, redefine them on different volumes before using the following procedures.

When recataloging non-VSAM data sets that reside on the system residence volume and that are not APF-authorized, consider specifying the VOLUMES and DEVICETYPES parameters using indirect volume serial number and device type (see "Defining the Non-VSAM System Data Sets" on page 4-5).

The following procedure deletes the SYSCATLG member (STEPA), then adds the new member, which will contain the volume serial number of the volume on which the production master catalog for the old production system resides (STEPB).

Step 1: Update the SYSCATLG member of SYS1.NUCLEUS of the newly generated system to contain the volume serial number of the volume that contains the old production system's master catalog, which resides on the 666666 volume.

```
//STEPA    EXEC PGM=IEHPROGM
//NEWRES   DD   DISP=OLD,UNIT=3380,VOL=SER=111111
//SYSPRINT DD SYSOUT=A
//SYSIN    DD   *
   SCRATCH DSNAME=SYS1.NUCLEUS,VOL=3380=111111,MEMBER=SYSCATLG
/*

//STEPB    EXEC PGM=IEBGENER
//SYSIN    DD   DUMMY
//SYSUT2   DD   DSN=SYS1.NUCLEUS(SYSCATLG),DISP=(MOD,KEEP),
//            DCB=(BLKSIZE=18432),VOL=SER=111111,UNIT=3380
//SYSPRINT DD SYSOUT=A
//SYSUT1   DD   *
666666     (See note below.)
/*
```

The input statement format for STEPB follows.

Note: Installations running MVS/DFP Version 3, when it is available, should use the following:

Column	*Content*
1–6	Volume serial number for the volume containing the master catalog
7	Type of catalog

Catalog	**Hex Code**	**Card Punch Code**
ICF	F1	1
ICF with SYS%	F2	2
SAM	00	12-0-9-8-1
	or	
	40	no punches

Note: Codes other than those specified are reserved. For information about SYS%, see *DFP: Managing Catalogs*.

8 Qualification levels of catalog aliases

Qualification level	**Hex Codes**
1	01 or F1 or 40
2	02 or F2
3	03 or F3
4	04 or F4

Hex Codes	**Card Punch Code**
01	12-1-9
02	12-2-9
03	12-3-9
04	12-4-9
40	no punches
F1	1
F2	2
F3	3
F4	4

Note: Codes other than those specified are reserved. For information about catalog aliases, see *DFP: Managing Catalogs*.

9–10 CAS service task lower limit value: C'18' to C'B4', where C'3C' is the default.

11–54 Master catalog name you specified in the old production system's sysgen, using the NAME parameter on the DATASET statement, or the NAME parameter of the access method services command DEFINE MASTERCATALOG or DEFINE USERCATALOG. You may also derive the master catalog name from the data component name printed using LISTCAT for an integrated catalog facility catalog, or the cluster name printed using LISTCAT for a VSAM catalog.

55–80 Unused

Note: Installations not running MVS/DFP Version 3 should use the following:

Column	*Content*
1–6	Volume serial number for the volume containing the master catalog
7	Type of catalog

Catalog	Hex Code	Card Punch Code
ICF	F1	1
VSAM	00	12-0-9-8-1
	or	
	40	no punches

Note: Codes other than those specified are reserved.

8	Unused; previously device type code
9–10	CAS service task lower limit value: C'18' to C'C8', where C'3C' is the default
11–54	Master catalog name you specified in the old production system's sysgen, using the NAME parameter on the DATASET statement, or the NAME parameter of the access method services command DEFINE MASTERCATALOG or DEFINE USERCATALOG. You may also derive the master catalog name from the data component name printed using LISTCAT for an integrated catalog facility catalog, or the cluster name printed using LISTCAT for a VSAM catalog.
55–80	Unused

Step 2: The volume that contains the master catalog, the page data sets, and the SYS1.STGINDEX data set of the newly generated system (the volume serial number is 555555) is no longer required. Demount it or vary it offline.

Step 3: Vary volumes 111111 and 333333 offline. Using the REFORMAT command of Device Support Facilities, change the volume serial numbers of these volumes, which contain the newly generated versions of SYS1.NUCLEUS, SYS1.LOGREC, SYS1.SVCLIB, SYS1.IMAGELIB, SYS1.TELCMLIB, SYS1.CMDLIB, etc. Change the volume serial numbers of 111111 and 333333 to the volume serial numbers of their counterparts on the old production system, 222222 and 444444.

Step 4: Recatalog any new system libraries. In this example, none are shown. However, if, for example, SYS1.CMDLIB were on the system residence volume of the newly generated system; but on some volume other than the system residence volume in the old production system, uncatalog it, then recatalog it using access method services.

Step 5: Remove from the system the old production system residence volume and the volume containing SYS1.IMAGELIB, etc., from the old production system (the volume serial numbers are 222222 and 444444).

Step 6: IPL the integrated system, using the new configuration.

Note: For information on creating an alternate master catalog, see *Catalog Administration Guide*.

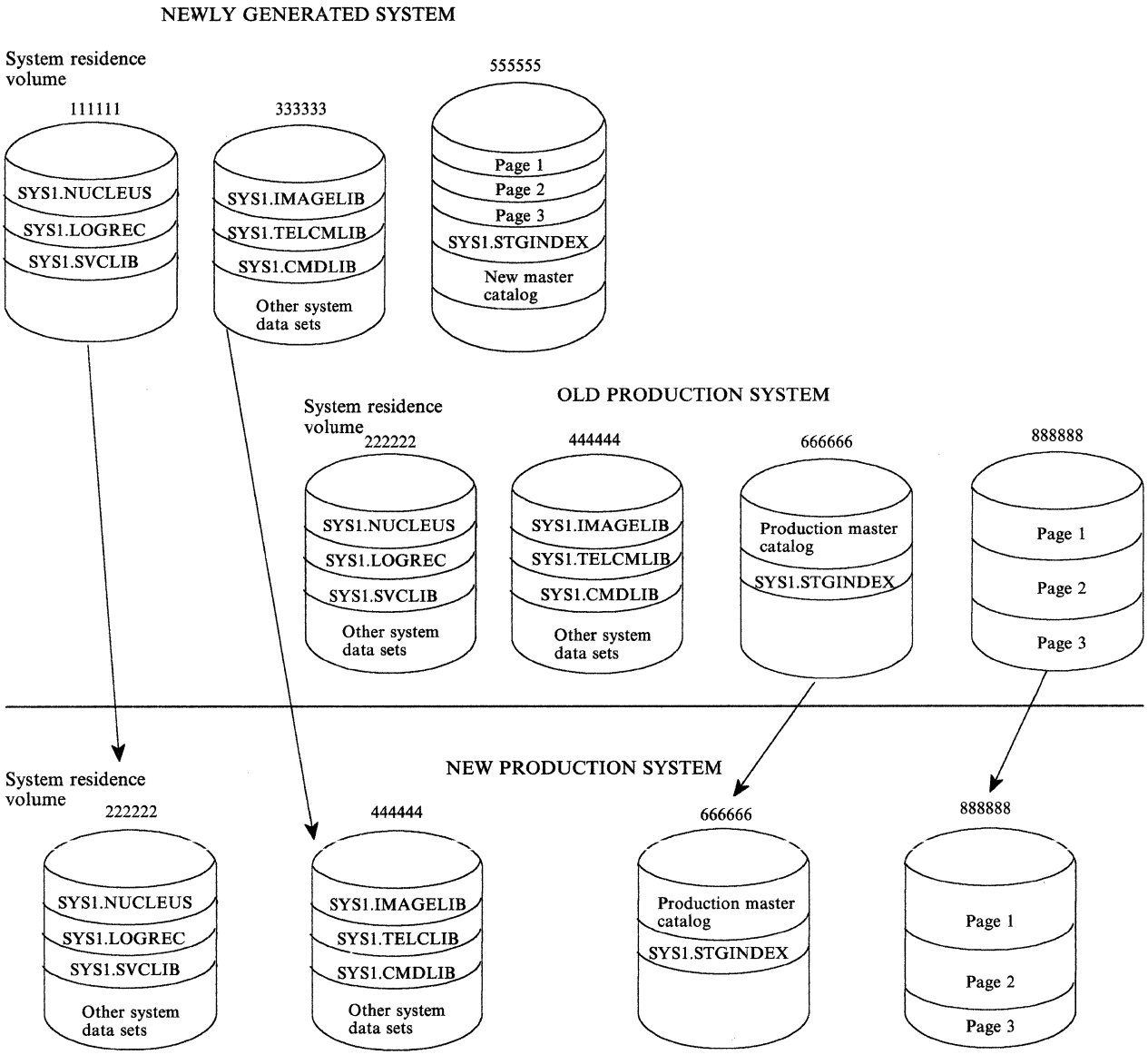


Figure 7-1. Reconfiguring the New Production System

Chapter 8. Installing ISMF

This chapter explains how to install the interactive storage management facility (ISMF) portion of DFP. Full ISMF function is contained in three products: DFP/ISMF, which is included on the DFP tape; Data Facility Data Set Services (DFDSS)/ISMF and Data Facility Hierarchical Storage Management (DFHSM)/ISMF, which are distributed separately. For information on how to install DFDSS/ISMF and DFHSM/ISMF, see the appropriate installation guides. For clarity, DFP/ISMF will be referred to as "ISMF" throughout this chapter.

Requirements

To use ISMF, you must have ISPF Version 2 Release 3 installed on your system. Figure 8-1 lists optional products you may want to install to provide additional function.

Figure 8-1. Products That Can Be Used with ISMF

Product	Required to
Resource Access Control Facility (RACF) Version 1 Release 1.8 and subsequent releases	Protect and report on ISMF
ISPF/PDF Version 2 Release 3, with PTF	Use BROWSE or EDIT
TSO/E Version 1 Release 4, with SPE	Use system-managed storage functions
TSO/E Version 2 Release 1	Use system-managed storage functions

Basic Tasks

ISMF is installed during normal sysgen processing; however, before users can use ISMF, you must:

1. Decide where to place the ISMF load modules
2. Decide whether ISMF will be protected by RACF
3. Make ISMF available to TSO/E users, either through a CLIST or logon JCL
4. Modify one or more menus to invoke ISMF

Considerations for ISMF Load Modules

This section discusses placement and authorization considerations for the ISMF load modules.

Placement Considerations

After you have completed sysgen processing, you have the following choices of where to place the load module libraries. The choice you make will affect the performance of your system. For example, there are maintenance considerations if the load modules exist anywhere other than in SYS1.DGTLLIB. In this case, you must notify System Modification Program / Extended (SMP/E) of the location of these modules.

You can choose from one of the following:

1. Leave SYS1.DGTLLIB where it is and add the data set name to the LOGON JCL, or to a new or existing CLIST for use with TSO.
2. Make the load modules a part of either of the following:
   - System link library
   - Pageable link pack area

Using the System Link Library: If you choose to use the system link library, you can do one of the following:

- Copy the data set to SYS1.LINKLIB.
- Add the data set name SYS1.DGTLLIB to LNKLSTnn, which is a member of the SYS1.PARMLIB.
- Copy the SYS1.DGTLLIB data set to an existing data set whose name is already in LNKLSTnn.

Using the Pageable Link Pack Area: If you choose to use the pageable link pack area, you can do one of the following:

- Copy the SYS1.DGTLLIB data set to SYS1.LPALIB.
- Add the SYS1.DGTLLIB data set name to LPALSTnn.
- Copy the SYS1.DGTLLIB data set to an existing data set that is part of LPALSTnn.

If you are not using SYS1.DGTLLIB at execution time and you have backed up and copied all of the load modules, you can delete the SYS1.DGTLLIB data set.

Authorization Considerations

You can use RACF to protect and report on ISMF. Using the program control feature, you can set up authorization levels for either all ISMF functions or for individual line operators and commands.

Because program control allows you to determine who can execute which ISMF function, you can use it to set up an authorization scheme that applies to both individual users and user groups. If you plan to use RACF to restrict ISMF functions, you must make the load module library a part of the system link library.

In addition to authorizing ISMF functions, you can use standard RACF authorization checking to limit access to individual data sets, volumes, or catalogs. Used in conjunction with program control, authorization checking will ensure that appropriate ISMF data and function are available to the users when needed. For detailed information on how to restrict ISMF functions, refer to the *ISMF User's Guide*.

Making ISMF Available to TSO/E Users

This section tells how to set up the execution environment to make ISMF available to TSO/E users.

Before TSO/E users can invoke ISMF, they must allocate specific data sets. Before this is possible, you need to make the execution libraries available either by modifying the TSO LOGON procedure or by setting up a CLIST.

Modifying the TSO LOGON Procedure

The following sections describe the statements you should include in a logon procedure for ISMF.

Specifying ISMF Execution Data Sets

The LOGON procedure should include the DD statements shown in Figure 8-2 to allocate the data sets required for ISMF operation. You can add the execution data sets for ISMF to the existing DD statements for each library; however, you must put them above the ISMF and ISPF data set names in the concatenation.

Note: These data sets must be allocated prior to invoking ISMF.

```
//STEPLIB   DD DSN=SYS1.DGTLLIB,DISP=SHR,
//          DD DSN=ISR.V2R2M0.ISRLOAD,DISP=SHR,
//          DD DSN=ISP.V2R2M0.ISPLOAD,DISP=SHR,
            .
            .
            .
//ISPPLIB   DD DSN=SYS1.DGTPLIB,DISP=SHR,
//          DD DSN=ISR.V2R2M0.ISRPLIB,DISP=SHR,
//          DD DSN=ISP.V2R2M0.ISPPLIB,DISP=SHR,
            .
            .
            .
//ISPMLIB   DD DSN=SYS1.DGTMLIB,DISP=SHR,
//          DD DSN=ISR.V2R2M0.ISRMLIB,DISP=SHR,
//          DD DSN=ISP.V2R2M0.ISPMLIB,DISP=SHR,
            .
            .
            .
//ISPTLIB   DD DSN=SYS1.DGTTLIB,DISP=SHR,
//          DD DSN=ISR.V2R2M0.ISRTLIB,DISP=SHR,
//          DD DSN=ISP.V2R2M0.ISPTLIB,DISP=SHR,
            .
            .
            .
```

Figure 8-2. Specifying ISMF Execution Data Sets

Setting Up a CLIST

As an alternative or supplement to modifying LOGON JCL, you can write a CLIST, which consists of a series of ALLOCATE commands for each execution library. You can add the name of the CLIST data set to the concatenation of SYSPROC or to the DDNAME ISPCLIB.

Figure 8-3 shows the ALLOCATE statements for each of the libraries.

```
ALLOCATE FILE(ISPLLIB) SHARE REUSE
                       DATASET('SYS1.DGTLLIB',       -
                               'ISR.V2R2M0.ISRLLIB',       -
                               'ISP.V2R2M0.ISPLLIB')
.
.
.
ALLOCATE FILE(ISPPLIB) SHARE REUSE                   -
                       DATASET('SYS1.DGTPLIB',       -
                               'ISR.V2R2M0.ISRPLIB',       -
                               'ISP.V2R2M0.ISPPLIB')
.
.
.
ALLOCATE FILE(ISPMLIB) SHARE REUSE                   -
                       DATASET('SYS1.DGTMLIB',       -
                               'ISR.V2R2M0.ISRMLIB',       -
                               'ISP.V2R2M0.ISPMLIB')
.
.
.
ALLOCATE FILE(ISPTLIB) SHARE REUSE                   -
                       DATASET('SYS1.DGTTLIB',       -
                               'ISR.V2R2M0.ISRTLIB',       -
                               'ISP.V2R2M0.ISPTLIB')
.
.
.
```

Figure 8-3. Setting Up a CLIST

Considerations for ISPTLIB and ISPTABL

After you allocate the execution data sets, you must decide whether users will share a common table library (SYS1.DGTTLIB), or whether each user will have a separate table library identified by his or her userid (userid.DGTTLIB).

This is an important consideration because ISMF saves the values that were last entered in the table. If each user has a separate table library, the user will see the values they last entered. However, if users share a table library, the values entered by the last user are the ones that are saved. The user may or may not see the values that they last entered.

Both ways of setting up the table library are shown below.

The CLIST in Figure 8-3 on page 8-4 sets up a common table library by allocating SYS1.DGTTLIB.

If you decide to allocate a separate table library for each user, do the following:

1. For each user who will have a separate table library, preallocate a data set called userid.DGTTLIB. The userid must follow the TSO naming conventions for data sets. userid.DGTTLIB must have the same attributes as SYS1.DGTTLIB.
2. Modify the CLIST in Figure 8-3 on page 8-4.
   a. Concatenate userid.DGTTLIB ahead of SYS1.DGTTLIB as shown in Figure 8-4. The userid is automatically appended to the data set name.

```
ALLOCATE FILE(ISPTLIB) SHARE REUSE                          -
                       DATASET(DGTTLIB,                     -
                               'SYS1.DGTTLIB',              -
                               'ISR.V2R2M0.ISRTLIB',        -
                               'ISP.V2R2M0.ISPTLIB')
```

Figure 8-4. Concatenating Separate Table Libraries

   b. Immediately following the statements in Figure 8-4, add an ALLOCATE statement for ISPTABL, the ISPF output data set:

```
ALLOCATE FILE(ISPTABL) SHARE REUSE                          -
                       DATASET(DGTTLIB)
```

Figure 8-5. CLIST Statements for a Separate Table Library

The CLIST has now been modified to include a separate table library for this user.

Modifying Menus to Invoke ISMF

The next installation step is to modify an interactive menu to invoke ISMF. You can do this by creating an option on one of the following menus:

- The ISPF/PDF Primary Option Menu
- The SPF Master Application Menu
- A menu of your own choosing

In the following example, we will create an option to invoke ISMF on the ISPF/PDF Primary Option Menu.

1. Log on to TSO. If you have coded your CLIST or logon JCL correctly, the ISPF/PDF Primary Option Menu automatically appears on your screen. Your screen should look similar to the one shown in Figure 8-6 on page 8-6.

```
                        ISPF/PDF PRIMARY OPTION MENU
OPTION  ===>
                                                                USERID   - L841557
   0   ISPF PARMS  - Specify terminal and user parameters     TIME     - 14:49
   1   BROWSE      - Display source data or output listings   TERMINAL - 3278
   2   EDIT        - Create or change source data             PF KEYS  - 24
   3   UTILITIES   - Perform utility functions
       .
       .
       .
   C   CHANGES     - Display summary of changes for this release
   T   TUTORIAL    - Display information about ISPF/PDF
   X   EXIT        - Terminate ISPF using log and list defaults
```

Figure 8-6. ISPF/PDF Primary Option Menu

2. Select option 3, Utilities.
3. The Utilities Selection Menu appears on your screen. Select option 3, move/copy.
4. Copy member ISR@PRIM from the data set 'ISR.V2R2M0.ISRPLIB' to 'SYS1.DGTPLIB'.
5. Press PF3 to return to the Primary Option Menu.
6. Select option 2, Edit.
7. When you are prompted for the name of the data set to edit, specify 'SYS1.DGTPLIB(ISR@PRIM)'
8. Your screen should now look similar to the one shown in Figure 8-7. Figure 8-7 and Figure 8-8 on page 8-7 show the source code for ISR@PRIM. Figure 8-7 shows the first section of the menu, the BODY section. The BODY section lists the options the user will see.

```
%----------------------  ISPF/PDF PRIMARY OPTION MENU  ----------------------
%OPTION  ===>_ZCMD
%                                                          +USERID   - &ZUSER
%   0 +ISPF PARMS  - Specify terminal and user parameters  +TIME     - &ZTIME
%   1 +BROWSE      - Display source data or output listings +TERMINAL - &ZTERM
%   2 +EDIT        - Create or change source data          +PF KEYS  - &ZKEYS
%   3 +UTILITIES   - Perform utility functions
    .
    .
    .
%   C +CHANGES     - Display summary of changes for this release
%   T +TUTORIAL    - Display information about ISPF/PDF
%   X +EXIT        - Terminate ISPF using log and list defaults
+Enter%END+command to terminate ISPF.
%
%
%
```

Figure 8-7. ISPF/PDF Primary Option Menu: Body Section

Figure 8-8 on page 8-7 shows the INIT and PROC sections. The INIT section specifies the processing that occurs before the panel is displayed. The PROC section contains the code to execute the options. You will be adding entries to the BODY and PROC sections.

```
)INIT
  .HELP = ISR00003
  &ZPRIM = YES         /* ALWAYS A PRIMARY OPTION MENU      */
  &ZHTOP = ISR00003    /* TUTORIAL TABLE OF CONTENTS        */
  &ZHINDEX = ISR91000  /* TUTORIAL INDEX - 1ST PAGE         */
  VPUT (ZHTOP,ZHINDEX)  PROFILE
)PROC
  &ZSEL = TRANS ( TRUNC ( &ZCMD,'.')
                0,'PANEL(ISPOPTA)'
                1,'PGM(ISRBRO) PARM(ISRBR001)'
                2,'PGM(ISREDIT) PARM(P,ISREDM01)'
                3,'PANEL(ISRUTIL)'
                .
                .
                .
                C,'PGM(ISPTUTOR) PARM(ISR00005)'
                T,'PGM(ISPTUTOR) PARM(ISR00000)'
              ' ',' '
                X,'EXIT'
                *,'?' )
  &ZTRAIL =  .TRAIL
)END
```

Figure 8-8. ISPF/PDF Primary Option Menu: INIT and PROC Sections

9. Move down the menu until you find option C, "Display summary of changes for this release".
10. Create a blank line below this option.
11. Enter the option letter and text for invoking ISMF. The option letter and the text you should enter are shown below:

```
%   I +ISMF      - Invoke interactive storage management facility
```

12. Move down the menu to the PROC section, and find the line of code that corresponds to option C:

```
C,'PGM(ISPTUTOR) PARM(ISR00005)'
```

13. Create a blank line below this line.
14. Enter the code to invoke ISMF. The text you should enter is shown below:

```
I,'PGM(DGTFMD01) PARM(&ZCMD) NEWAPPL(DGT ) NOCHECK'
```

    Now, when you select option I from ISPF/PDF Primary Option Menu, the ISMF Primary Option Menu is displayed.
15. Press PF3 to save the primary option menu.

Testing ISMF

After you have modified a menu to invoke ISMF, you can test ISMF by following the steps in "Testing ISMF" on page 6-2.

Appendix A. Installing the DFP Starter Set

This appendix explains how to install the Data Facility Product (DFP) Starter Set. For details on the content of the Starter Set, see *Storage Administration Reference*.

Requirements

To use the starter set, you must have MVS/DFP Version 3 installed on your system and the supporting levels of required and corequisite products or their equivalents.

Basic Tasks

To install the starter set, run the IMPORT job that resides in SYS1.SAMPLIB.

Importing the Starter Set

The JCL to import the Starter Set resides in SYS1.SAMPLIB, member name DFPSSIMP. After you run the DFPSSIMP JCL, the starter set is ready to use. See *Storage Administration Reference* for ideas on how to customize and use the Starter Set.

Appendix B. Sysgen Example

The following example illustrates Stage 1 statements for a JES2 sysgen.

```
         TITLE 'SAMPLE MVS/XA SYSGEN'
*           DATA MANAGEMENT OPTIONS
DATAMAN  DATAMGT   ACSMETH=(BTAM,TCAM,ISAM),GRAPHCS=(PORRTNS,GSP)
*           SYSTEM DATA SET SPECIFICATIONS
BRODCAST DATASET   BRODCAST,VOL=(SG2001,3380),SPACE=(CYL,(3))
CMDLIB   DATASET   CMDLIB,VOL=(SG2001,3380),SPACE=(CYL,(6,2,100))
DAE      DATASET   DAE,VOL=(SG2001,3380),SPACE=(TRK,(6,2))
DGTCLIB  DATASET   DGTCLIB,VOL=(SG2001,3380),SPACE=(CYL,(2,1,20))
DGTLLIB  DATASET   DGTLLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTMLIB  DATASET   DGTMLIB,VOL=(SG2001,3380),SPACE=(CYL,(1,1,20))
DGTPLIB  DATASET   DGTPLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,1,50))
DGTSLIB  DATASET   DGTSLIB,VOL=(SG2001,3380),SPACE=(CYL,2,1,20))
DGTTLIB  DATASET   DGTTLIB,VOL=(SG2001,3380),SPACE=(TRK,(9,1,10))
DUMPDS1  DATASET   DUMP01,VOL=(SG2001,3380),SPACE=(CYL,(3))
DUMPDS2  DATASET   DUMP02,VOL=(SG2001,3380),SPACE=(CYL,(3))
HELP     DATASET   HELP,VOL=(SG2001,3380),SPACE=(CYL,(5,1,15))
IMAGELIB DATASET   IMAGELIB,VOL=(SG2001,3380),SPACE=(CYL,(10,3,100))
INDMAC   DATASET   INDMAC,VOL=(PUGRE1,3380),SPACE=(CYL,(45,3,5))
ISAMLPA  DATASET   ISAMLPA,VOL=(SG2001,3380),SPACE=(CYL,(5,1,30))
LINKLIB  DATASET   LINKLIB,VOL=(SG2001,3380),SPACE=(CYL,(60,9,600))
LOGREC   DATASET   LOGREC,VOL=(SG2001,3380),SPACE=(CYL,(5))
LPALIB   DATASET   LPALIB,VOL=(SG2001,3380),SPACE=(CYL,(24,3,350))
MACLIB   DATASET   MACLIB,VOL=(SG2001,3380),SPACE=(CYL,(55,5,150))
MANX     DATASET   MAN7,VOL=(SG2001,3380),SPACE=(CYL,(30))
MANY     DATASET   MANY,VOL=(SG2001,3380),SPACE=(CYL,(30))
MIGLIB   DATASET   MIGLIB,VOL=(SG2001,3380),SPACE=(CYL,(60,9,600))
NUCLEUS  DATASET   NUCLEUS,VOL=(SG2001,3380),SPACE=(CYL,(35,,100))
PARMLIB  DATASET   PARMLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,,25))
PROCLIB  DATASET   PROCLIB,VOL=(SG2001,3380),SPACE=(CYL,(3,5,200))
SBLSCLI0 DATASET   SBLSCLI0,VOL=(SG2001,3380),SPACE=(TRK,(4,4,5))
SBLSKEL0 DATASET   SBLSKEL0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
SBLSMSG0 DATASET   SBLSMSG0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
SBLSPNL0 DATASET   SBLSPNL0,VOL=(SG2001,3380),SPACE=(CYL,(2,1,15))
SBLSTBL0 DATASET   SBLSTBL0,VOL=(SG2001,3380),SPACE=(TRK,(2,1,5))
SMFDS    DATASET   SMFDSN=(1,2),VOL=(SG2001,3380),SPACE=(CYL,(30))
STGINDEX DATASET   STGINDEX,VOL=(SG2001,3380),SPACE=(CYL,(12))
SVCLIB   DATASET   SVCLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,3,150))
TCOMMAC  DATASET   TCOMMAC,VOL=(PUGRE3,3380),SPACE=(CYL,(5,1,50))
TELCMLIB DATASET   TELCMLIB,VOL=(SG2001,3380),SPACE=(CYL,(5,2,70))
UADS     DATASET   UADS,VOL=(SG2001,3380),SPACE=(CYL,(5,1,20))
VTAMLIB  DATASET   VTAMLIB,VOL=(PUGRE2,3380),SPACE=(CYL,(5,2,50))
*           THE MASTER CATALOG
ICFMASTR DATASET   ICFCATLG,VOL=(SG2001,3380),SPACE=(CYL,(30,3)),          X
               NAME=SYS1.ICFCAT
SAMPLIB  DATASET   SAMPLIB,VOL=(SG2001,3380),SPACE=(CYL,(10,5,50))
PAGE1    DATASET   PAGEDSN=SG2PAGE1,VOL=(SG2001,3380),SPACE=(CYL,(15))
PAGE2    DATASET   PAGEDSN=SG2PAGE2,VOL=(SG2001,3380),SPACE=(CYL,(75))
PAGE3    DATASET   PAGEDSN=SG2PAGE3,VOL=(SG2001,3380),SPACE=(CYL,(100))
SWAPDS1  DATASET   SWAPDSN=SG2SWAP1,VOL=(SG2001,3380),SPACE=(CYL,(40))
SWAPDS2  DATASET   SWAPDSN=SG2SWAP2,VOL=(SG2001,3380),SPACE=(CYL,(40))
SWAPDS3  DATASET   SWAPDSN=SG2SWAP3,VOL=(SG2001,3380),SPACE=(CYL,(40))
*           THIS STATEMENT SPECIFIES A SYSGEN
         GENERATE  RESVOL=(SG2001,3380),INDEX=SG2,JCLASS=K,OCLASS=K
         END
```

Figure B-1. Example Stage I Input for a JES2 Sysgen

Appendix C. Making ISAM Available on a System with DFP 3.1.0 Installed

This appendix explains how to make the index sequential access method (ISAM) available on a system with DFP 3.1.0 installed. The ISAM component modules reside in a library called SYS1.ISAMLPA, not in SYS1.LPALIB. This placement makes it possible to include or exclude the access method at system IPL time instead of system generation time. The following steps are required to make ISAM available:

1. Add SYS1.ISAMLPA to the LPALSTyy member in parmlib.
2. Add SYS1.ISAMLPA to the IEAAPFzz member in parmlib.
3. Update the IEASYxx member to point to updated LPALSTyy and IEAAPFzz members:

   APF=zz

   LPALST=(yy,L)
4. Catalog SYS1.ISAMLPA in the master catalog for the system.
5. IPL system with SYSP=xx and CLPA options.

Index

E

G

H

I

J

L

M

T

U

V

W

Readers' Comments

MVS/ESA
System Generation

MVS/System Product:
JES2 Version 3
JES3 Version 3

Publication No. GC28-1825-3

Use this form to tell us what you think about this manual. If you have found errors in it, or if you want to express your opinion about it (such as organization, subject matter, appearance) or make suggestions for improvement, this is the form to use.

To request additional publications, or to ask questions or make comments about the functions of IBM products or systems, you should talk to your IBM representative or to your IBM authorized remarketer. This form is provided for comments about the information in this manual and the way it is presented.

When you send comments to IBM, you grant IBM a nonexclusive right to use or distribute your comments in any way it believes appropriate without incurring any obligation to you.

Be sure to print your name and address below if you would like a reply.

Name

Address

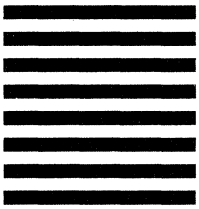
Company or Organization

Phone No.

Cut or Fo
Along Lin

Fold and Tape **Please do not staple** Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES



BUSINESS REPLY MAIL

FIRST CLASS MAIL PERMIT NO. 40 ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corporation
Department D58, Building 921-2
PO BOX 950
POUGHKEEPSIE NY 12602-9935

Fold and Tape **Please do not staple** Fold and Tape

Cut or F
Along L

Program Number
5685-001
5685-002

File Number
S370-34

GC28-1825-3

Printed in U.S.A.