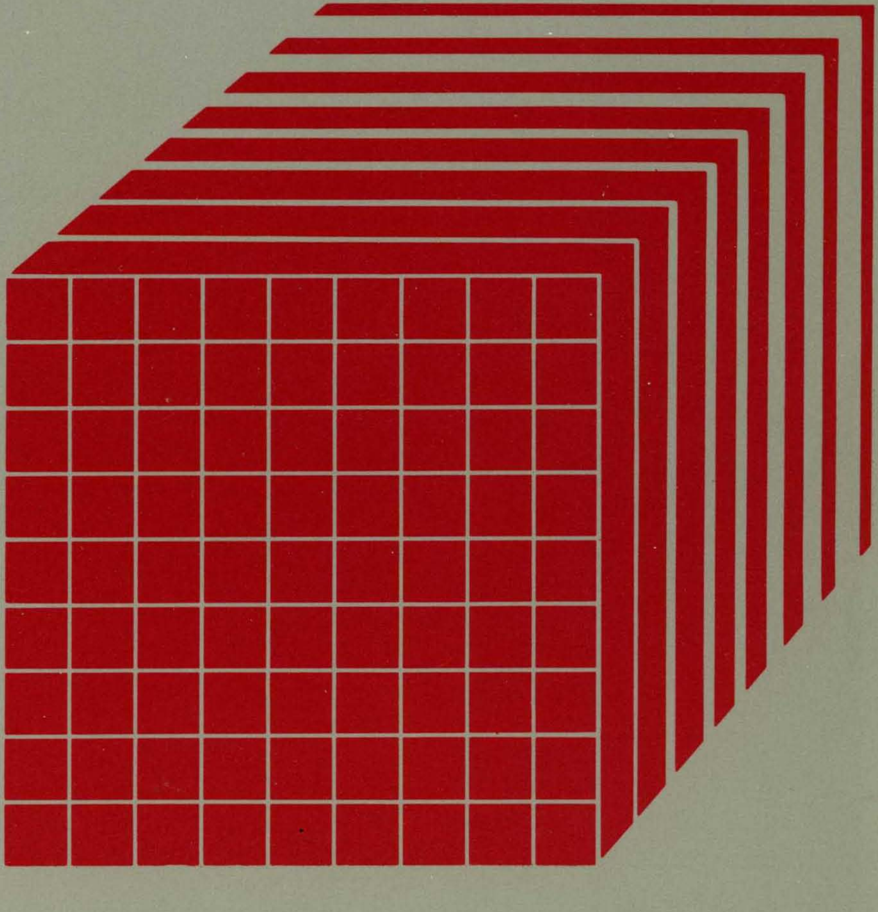
Virtual Machine/ System Product

Installation Guide

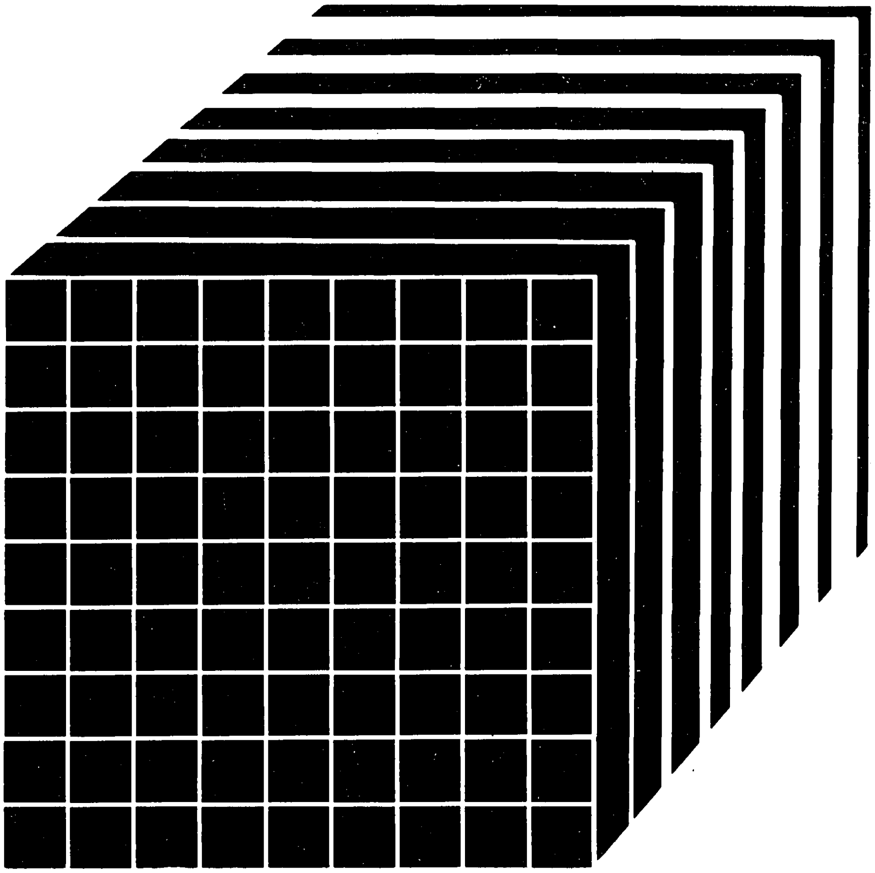
Release 3.1



Virtual Machine/
System Product

Installation Guide

Release 3.1



Second Edition (February 1984)

This edition, SC24-5237-1, is a major revision of SC24-5237-0, and applies to Release 3, modification level 1 (Release 3.1) Virtual Machine/System Product (VM/SP), program number 5664-167, and to all subsequent releases and modifications until otherwise indicated in new editions or Technical Newsletters.

Summary of Changes

For a list of changes, see page iii.

Technical changes and additions to text and illustrations are indicated by a vertical line to the left of the change.

The Appendixes J through P contain listings of the sample files used with this release. The changes in the sample file listings are not identified by the vertical line.

Changes are made periodically to the information contained herein; before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370 and 4300 Processors Bibliography*, GC20-0001, for the editions that are applicable and current.

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 program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Programming Publications, Department G60, P.O. Box 6, Endicott, NY, U.S.A. 13760. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Summary of Changes

Summary of Changes for SC24-5237-1 for VM/SP Release 3.1

Optional Feature Program Product Installation

New: Chapter 6 and Appendix I contain information for installing the optional feature program products.

Three new EXECs are added for this release. They are: PPPREP, INSTFPP, and DISKMAP.

Changed: The GENERATE EXEC is changed for this release.

Installation Guide Reorganization

Changed: The sample files are in Appendixes J through P. Chapter numbers after chapter 5 have been changed because of the addition of Chapter 6 and the relocation of the sample files.

Miscellaneous

Various minor technical and editorial changes have been made throughout the publication.

Summary of Changes for SC24-5237-0 for VM/SP Release 3

- The CMSSEG Saved Segment has been removed. Code from CMSSEG has been incorporated into the CMS Nucleus.
- The CMSXGEN Procedure which formerly generated CMSSEG has been deleted.
- EREP files have been removed from the Product Tape.
- Space on the 190 minidisk has been reserved for the CMSL Nucleus.

The Sources for the SC24-5237-0 edition of this Book

This book includes material that originally appeared in the *VM/SP Planning and System Generation Guide*. (The *VM/SP: Planning and System Generation Guide* has been restructured to include planning information only. It has been retitled *VM/SP Planning Guide and Reference.*)

Most of this book came from Parts 3 through 5 and Appendixes C, F, and G of the *VM/SP Planning and System Generation Guide.*

- Chapters 1 through 5 of this book were taken from "Part 3. Generating VM/SP (CP and CMS)" of the *VM/SP Planning and System Generation Guide* and "Chapter 4. Installation Procedures Using the IPO1 Tape" of the *Universal Program Directory for VM/SP System IPO/E Release 2.1.*

- Chapters 6 through 12 of this book were created from sample files previously listed in "Chapter 23. Creating Your VM/SP Directory," "Chapter 24. Preparing System Name Table File (DMKSNT)," and "Chapter 27. Sample System Generation Using a Starter System" of the *VM/SP Planning and System Generation Guide* and from Appendixes A through F of the *VM/System Product System Installation Productivity Option/Extended: Planning Guide.*

- Chapters 13 through 16 of this book were taken from "Part 5. Updating VM/SP" of the *VM/SP Planning and System Generation Guide.*

- Appendix A of this book was taken from "Appendix C. CP/CMS Nucleus/Module Regeneration Requirements" of the *VM/SP Planning and System Generation Guide.*

- Appendix B of this book was taken from "Appendix F. A Sample EXEC Procedure for Copying VSE Macros into a CMS MACLIB" of the *VM/SP Planning and System Generation Guide.*

- Appendix C of this book was taken from "Appendix G. Generating VM/SP Without a VM/SP Starter System or the Merged Product Tapes" of the *VM/SP Planning and System Generation Guide.*

- Appendix D of this book was taken from "Chapter 33. Generating and Loading the 3704/3705 Control Program" of the *VM/SP Planning and System Generation Guide.*

- Appendix E of this book was taken from "Chapter 27. Sample System Generation Using a Starter System" of the *VM/SP Planning and System Generation Guide.*

- Appendix F of this book has been added as a result of an authorized program analysis report (APAR).

- Minor technical and editorial changes are also included.

Preface

The Intent of This Book

The *VM/SP Installation Guide* discusses how to install and service IBM Virtual Machine System Product (VM/SP). The necessary material, procedures, and examples are described in this book.

A general understanding of System/370 data processing and teleprocessing techniques is assumed. Also, a review of the contents of *VM/SP Introduction*, *VM/SP Planning Guide and Reference*, and *VM/SP System Product Editor User's Guide* is required before this book is used.

This book has two parts, plus appendixes.

"Part 1. VM/SP Installation" describes the step-by-step procedure for installing CP and CMS. Two separate procedures are presented. The first is intended for those of you who wish to install VM/SP using both the Starter System and Product Tapes. The other is meant to be used by current VM/SP users who wish to install only the Product Tape using their existing VM/SP system.

Part 1 also includes a procedure for verifying CP and CMS, a method to load and save saved segments, and the procedure for installing the optional feature program products.

"Part 2. VM/SP Service" describes the procedures, programs, and EXECs used to update VM/SP source code and macro libraries.

Also included are appendixes about:

- CP/CMS nucleus/module regeneration requirements
- A sample EXEC procedure to copy VSE macros into a CMS MACLIB
- Installing VM/SP without both distribution tapes
- Generating the 3704/3705 control program
- Alternate CMS nucleus placement
- Creating additional CMS segment containing Y-stats
- Special options for CP
- Recording references for installation.
- A list of the program products contained on the Feature tape(s) and considerations for installing the program products.
- Sample file listings of configurations by device type.

In this book, the following terms have extended meanings:

- The term "Merged Product Tape" or just "Product Tape" refers to the tape which contains VM/SP and VM/370 Release 6 merged. It contains all of the CP, CMS and HELP files necessary to generate a complete VM/SP system including those required from VM/370 Release 6.

- The term "VM/SP Only Product Tape" refers to the tape which contains VM/SP modules only. This tape contains only the CP and CMS modules and macros that have been changed or added since VM/370 Release 6. (Often it is referred to as an unmerged product tape.)

- The term "3330 series" refers to the IBM 3330 Disk Storage Models 1 and 11; and the IBM 3333 Disk Storage and Control, Models 1 and 11.

- The term "3340 series" refers to the IBM 3340 Disk Storage, Models A2, B1 and B2, and the 3344 Direct Access Storage Model B2.

- The term "3350 series" refers to the IBM 3350 Direct Access Storage Models A2 and B2 in native mode.

- The term "3375" refers to the IBM 3375 Direct Access Storage Device.

- The term "3380" refers to the IBM 3380 Direct Access Storage Device.

- The term "FB-512" refers to the IBM 3310 and 3370 Direct Access Storage Devices.

- The term "3705" refers to the IBM 3705-I and 3705-II Communications Controllers, unless otherwise specified.

- The term "3270" refers to all VM/SP supported virtual machine display consoles unless otherwise noted. A specific device type is used only when it is necessary to show that device types differ.

- Information about display terminal usage also applies to the IBM 3138, 3148 and 3158 Display Consoles in display mode, unless otherwise noted.

- The term "display device" refers to any VM/SP supported system console terminal that displays data on a screen.

- Unless otherwise noted, the term "ATTN key" implies "(or equivalent)." It refers to the "signal interrupt control key" as listed for each terminal or console type in the *VM/SP Terminal Reference.* (For example, on the 3278 terminal it is the PA1 key.)

- Unless otherwise noted, the term VSE refers to the DOS/VSE system control program and the VSE/Advanced Functions program products combined.

  In certain cases, the term DOS is still used as a generic term. For example, disk packs initialized for use with VSE or any prior DOS or DOS/VS system may be referred to as DOS disks.

  The DOS-like simulation environment provided under the CMS portion of VM/SP continues to be referred to as CMS/DOS.

- CMS/DOS is part of the CMS system and is not a separate system. The term "CMS/DOS" states concisely that the VSE simulation mode of CMS is currently active. In other words, the CMS command **set dos on** already has been invoked.

- The phrase "the CMS file system" refers to disk files that are in CMS's 1K, 2K, or 4K fixed physical block format. CMS's VSAM data sets are not included.

- The phrase "System Product Editor" implies the use of the XEDIT command, XEDIT subcommands and macros, and XEDIT prefix subcommands.

The Contents of This Book

This book contains descriptions of the magnetic tapes and disk packs required to generate VM/SP. Step-by-step procedures for using them are also included. The descriptions and procedures show you how to:

- Format the DASD needed to install a system
- Install a new VM/SP system using the Starter System Tape
- Install VM/SP when not using the Starter System Tape
- Tailor the new system to specific needs
- Check that the newly installed CP and CMS are working properly
- Display sample directories, product tape files, and pack layouts
- Load and save discontiguous saved segments
- Update VM/SP thorough recommended service procedures
- Use EXEC procedures and commands.

Whether you are a new or existing user of VM/SP, you will find this book an important installation aid. You may use this guide to install the system on several types of direct access storage devices (DASD). Included are instructions for using the following devices: 3310, 3330, 3340, 3350, 3370, 3375, and 3380.

Corequisite Publications

- For information about the VM/SP system, the following publications are corequisite:

  Virtual Machine/System Product:

  Introduction, GC19-6200
  Planning Guide and Reference, SC19-6201
  Operator's Guide, SC19-6202
  System Programmer's Guide, SC19-6203
  System Messages and Codes, SC19-6204
  Terminal Reference, GC19-6206
  CMS Command and Macro Reference, GC19-6209
  CMS User's Guide, GC19-6210
  CP Command Reference for General Users, GC19-6211
  System Product Editor Command and Macro Reference, SC24-5221
  System Product Interpreter Reference, SC24-5239

  References in the text to titles of VM/SP books are given in abbreviated form.

- For information regarding 3270 display systems, the following publication is corequisite:

  3270 Information Display System Library User's Guide, GA23-0058

- For information concerning SNA supported devices, refer to the following publication:

  VM/VCNA Installation, Operations, and Terminal Use, SC27-0502

- If the IBM 3850 Mass Storage System is attached to the VM/SP processor, the following publications are corequisite:

  IBM 3850 Mass Storage System (MSS) Installation Planning and Table Create, GC35-0028

- For information regarding 3704/3705 Communications Controllers, the following publications are corequisite:

  IBM 3704 and 3705 Control Program Generation and Utilities Guide and Reference Manual (OS/VS TCAM Levels 5 and 6 in VS1; VS2 rel 1.6, 1.7, 2, SCP 5744-BA1, GC30-3007

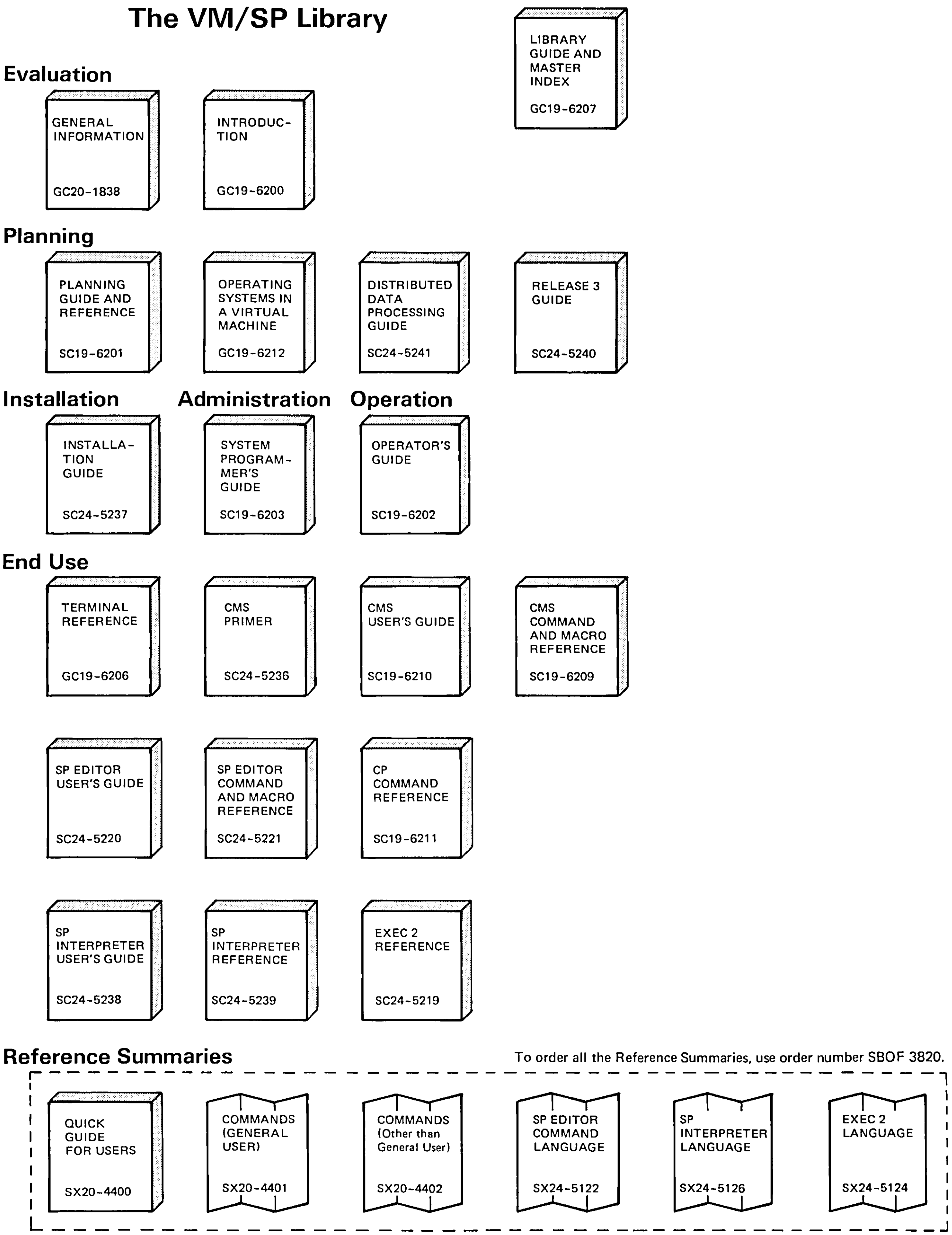
  IBM 3704 and 3705 Control Program Generation and Utilities Guide and Reference Manual (TCAM 10 SVS - 5742-017) SCP 5742, 5744-AN1/BA2, 5747-AG | -AG2, GC30-3008

  Guide to Using the IBM 3704 Communications Controller Control Panel, GA27-3086

  Guide to Using the IBM 3705 Control Panel, GA27-3087

- For information regarding DASD support, the following publication is corequisite:

  Device Support Facility User's Guide and Reference, GC35-0033


The VM/SP Library
LIBRARY GUIDE AND MASTER INDEX GC19-6207
Evaluation
GENERAL INFORMATION GC20-1838
INTRODUCTION GC19-6200
Planning
PLANNING GUIDE AND REFERENCE SC19-6201
OPERATING SYSTEMS IN A VIRTUAL MACHINE GC19-6212
DISTRIBUTED DATA PROCESSING GUIDE SC24-5241
RELEASE 3 GUIDE SC24-5240
Installation
INSTALLATION GUIDE SC24-5237
Administration
SYSTEM PROGRAMMER'S GUIDE SC19-6203
Operation
OPERATOR'S GUIDE SC19-6202
End Use
TERMINAL REFERENCE GC19-6206
CMS PRIMER SC24-5236
CMS USER'S GUIDE SC19-6210
CMS COMMAND AND MACRO REFERENCE SC19-6209
SP EDITOR USER'S GUIDE SC24-5220
SP EDITOR COMMAND AND MACRO REFERENCE SC24-5221
CP COMMAND REFERENCE SC19-6211
SP INTERPRETER USER'S GUIDE SC24-5238
SP INTERPRETER REFERENCE SC24-5239
EXEC 2 REFERENCE SC24-5219
Reference Summaries
QUICK GUIDE FOR USERS SX20-4400
COMMANDS (GENERAL USER) SX20-4401
COMMANDS (Other than General User) SX20-4402
SP EDITOR COMMAND LANGUAGE SX24-5122
SP INTERPRETER LANGUAGE SX24-5126
EXEC 2 LANGUAGE SX24-5124

To order all the Reference Summaries, use order number SBOF 3820.

Figure 1 (Part 1 of 2). Virtual Machine/System Product Library

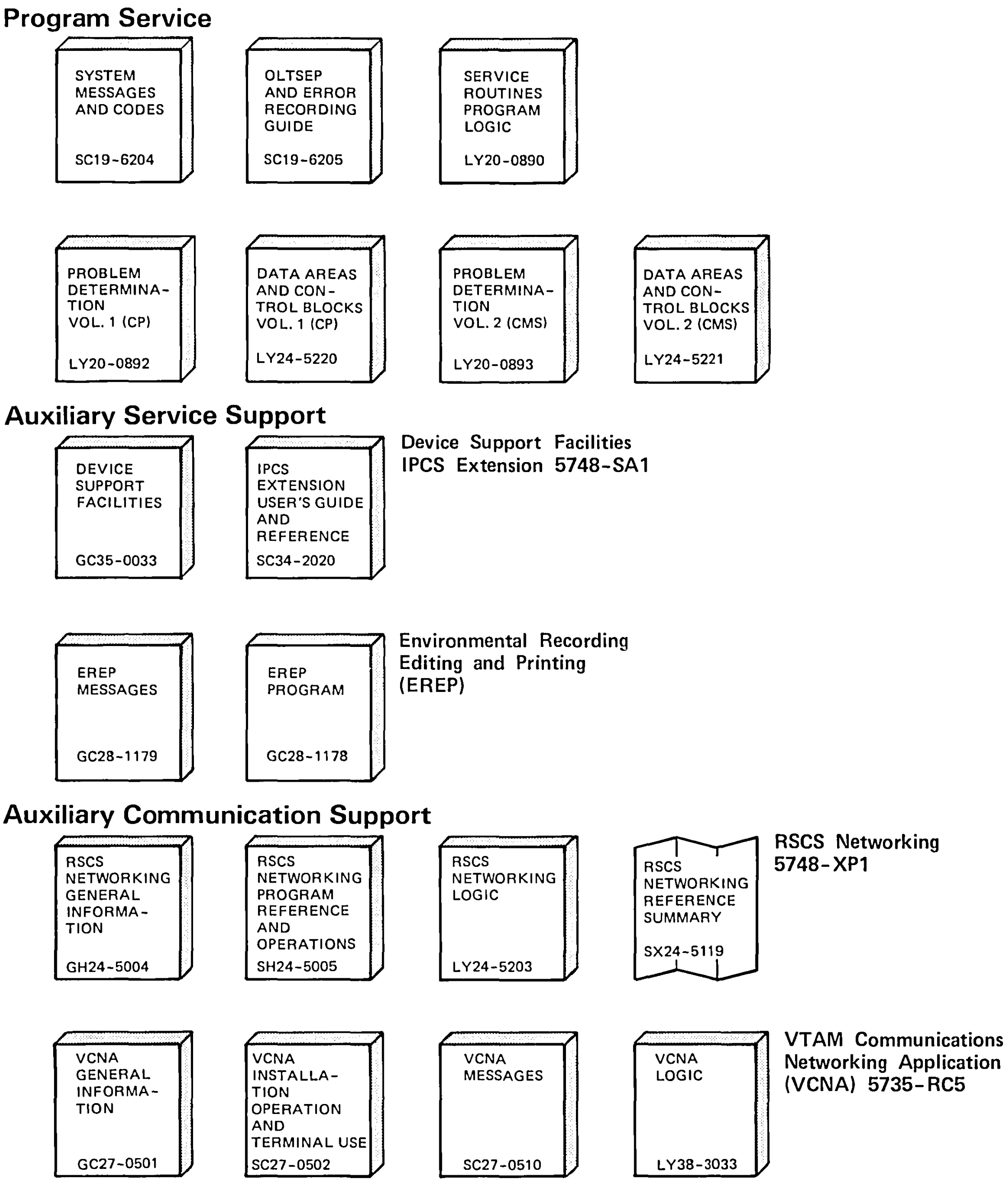

Program Service
SYSTEM MESSAGES AND CODES SC19-6204
OLTSEP AND ERROR RECORDING GUIDE SC19-6205
SERVICE ROUTINES PROGRAM LOGIC LY20-0890
PROBLEM DETERMINATION VOL. 1 (CP) LY20-0892
DATA AREAS AND CONTROL BLOCKS VOL. 1 (CP) LY24-5220
PROBLEM DETERMINATION VOL. 2 (CMS) LY20-0893
DATA AREAS AND CONTROL BLOCKS VOL. 2 (CMS) LY24-5221
Auxiliary Service Support
DEVICE SUPPORT FACILITIES GC35-0033
IPCS EXTENSION USER'S GUIDE AND REFERENCE SC34-2020
Device Support Facilities
IPCS Extension 5748-SA1
EREP MESSAGES GC28-1179
EREP PROGRAM GC28-1178
Environmental Recording Editing and Printing (EREP)
Auxiliary Communication Support
RSCS NETWORKING GENERAL INFORMATION GH24-5004
RSCS NETWORKING PROGRAM REFERENCE AND OPERATIONS SH24-5005
RSCS NETWORKING LOGIC LY24-5203
RSCS NETWORKING REFERENCE SUMMARY SX24-5119
RSCS Networking 5748-XP1
VCNA GENERAL INFORMATION GC27-0501
VCNA INSTALLATION OPERATION AND TERMINAL USE SC27-0502
VCNA MESSAGES SC27-0510
VCNA LOGIC LY38-3033
VTAM Communications Networking Application (VCNA) 5735-RC5

Figure 1 (Part 2 of 2). Virtual Machine/System Product Library

Contents

Figures

Part 1. VM/SP Installation

Throughout this book you will see references to three types of tapes. The tapes and the information they contain are:

Starter System Tape

The tape you start with when installing "from scratch." Starter system tapes are DASD-type specific; for instance, you can't use "3350 Starter System" on any other DASD-type than 3350. Once restored, the Starter System is a very basic VM system which you use to build your own production system.

Note: The Starter System is NOT a production system; its only use is installing the VM/SP Merged Product Tape.

VM/SP Product Tape

VM/SP is shipped in two different ways, depending on the feature codes you specify in your order:

"Merged Product Tape" contains all current VM code. It is "merged" because it includes both VM/SP and VM/370 release 6 modules and macros. Most of this book is devoted to installation using the VM/SP Merged Product Tape.

"VM/SP Product Tape" contains only those modules and macros added or altered since VM/370 release 6. Installation using this tape is detailed in Appendix C, "VM/SP Without the Starter System and the Merged Product Tapes" on page 251.

Optional Feature Products Tape

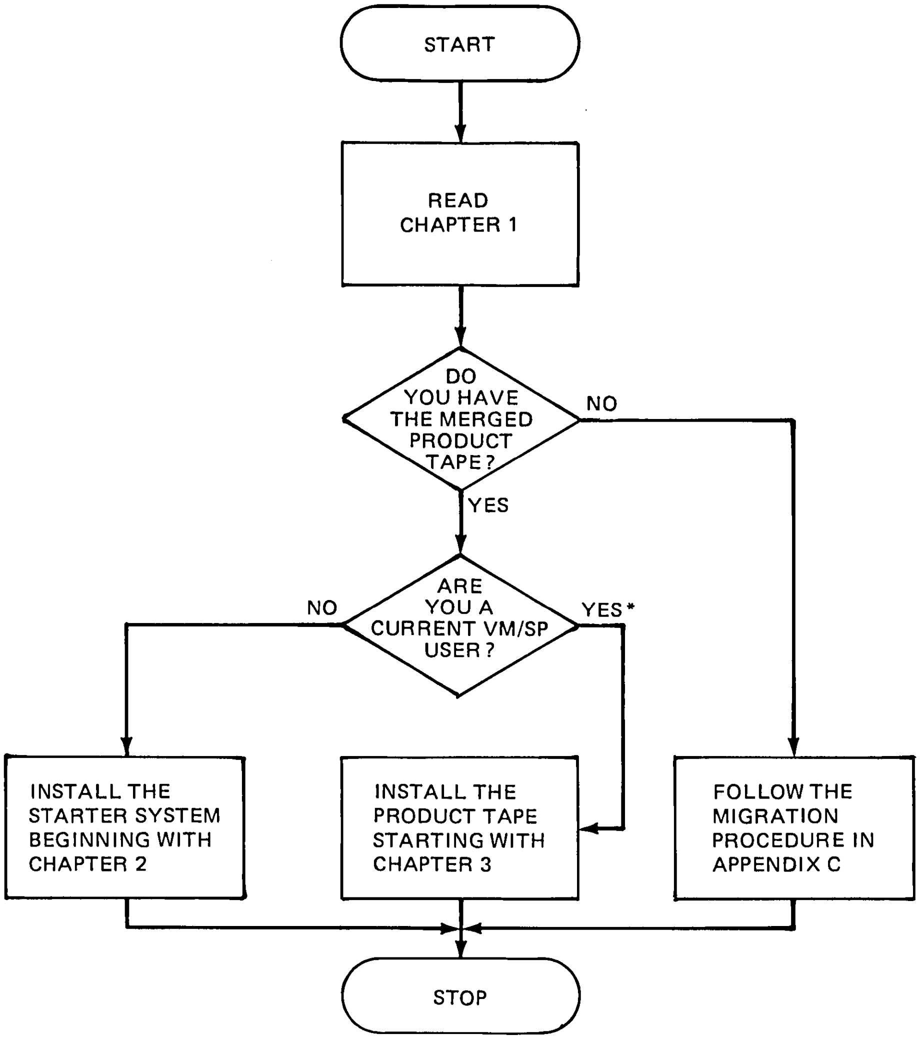
The tape that contains the Optional Feature Program Products that you will install with your system using INSTFPP.

Where to Start

As shown in Figure 2 on page 2, after all users read "Chapter 1. Introduction to VM/SP Installation" on page 5, new users of VM/SP with the Merged Product Tape begin with "Chapter 2. Installation When Using the Starter System Tape" on page 15. Existing VM/SP users with the Merged Product Tape proceed to "Chapter 3. Installation When Not Using the Starter System Tape" on page 77. Other migrating users proceed to Appendix C, "VM/SP Without the Starter System and the Merged Product Tapes" on page 251 for the VM/SP generation procedure.

At this time, you should make a copy of the chart in Appendix H, "Installation Reference Worksheet" on page 285 and fill in the blanks as the needed information becomes available. This volume number, address, label, and device type information will be needed during the installation of VM/SP.

Unless you are a current VM/SP user, you must install the Starter System before you attempt to install the Merged Product Tape. (If you are a current VM/SP user, you are also encouraged to install the Starter System for a much simpler procedure.) See Figure 2 on page 2 for the recommended installation procedure for your situation.


START
READ CHAPTER 1
DO YOU HAVE THE MERGED PRODUCT TAPE ?
NO
YES
ARE YOU A CURRENT VM/SP USER ?
NO
YES*
INSTALL THE STARTER SYSTEM BEGINNING WITH CHAPTER 2
INSTALL THE PRODUCT TAPE STARTING WITH CHAPTER 3
FOLLOW THE MIGRATION PROCEDURE IN APPENDIX C
STOP

*If you are a current VM/SP user and do not wish to migrate, you can answer "no" to this question. The "yes" path assumes a higher VM knowledge level.

Figure 2. Organization of VM/SP Installation. Where to start depending on user's system.

About Part 1

Chapter 1 is an introduction to the VM/SP installation process for all users. Pre-installation requirements and modifications are presented. Applicable starter systems are listed and the distribution tape layouts are described. An explanation of *first level* and *second level* system operation is included as well as how the terms ATTN and ENTER are used in this book. The last part of Chapter 1 shows the format procedure used throughout the book.

Chapters 2 and 3 describe the step-by-step generation procedures for CP and CMS (for both new and existing users with the Merged Product Tape). The Installation Verification Procedure (IVP), which checks that CP and CMS are working properly, is covered in Chapter 4 and the procedures for loading and saving discontiguous saved segments are in Chapter 5. Chapter 6 contains the procedures for installing the optional feature program products.

Use the following tab reference to help you locate areas of this book that pertain to your installation.

All VM/SP users begin here	A
Installing the Starter System and the Merged Product Tape	B
Installing the Merged Product Tape (Current VM/SP users only)	C
Installing the VM/SP Only Product Tape	D
Information about verifying test installation (IVP)	E
Information about loading and saving discontiguous saved segments	F
Information about installing optional feature program products	G

Chapter 1. Introduction to VM/SP Installation

Requirements to Satisfy Before Starting

Before you start to install VM/SP, have available:

- At least two physical (real) disk drives.
- At least one physical (real) tape drive.
- Two scratch disks, one for the Starter System, the other for work space and for system staging areas.
- Optional scratch disk:
  - Additional user space on (VMSEXT) on 3310, 3330 and 3340 systems.
  - Holding source code (VMSTGE) on all DASD systems.
  - Holding CMS source code (VMSTG2) on 3310 and 3340 systems. (The MAINT userid has an entry for minidisk 393 on VMSTG2. This volume must be formatted and labeled, using the format/allocate program, prior to loading source code.)
- At least one scratch tape.
- The Starter System Tape or an existing VM/SP system.
- The system Program Update Tape (PUT), if any. (With your initial VM/SP shipment, you might receive a VM/SP service tape in PUT format.)
- The stacked Optional Feature Program Product Tape(s).
- The EREP Program Product Tape if EREP is not on the Optional Feature Program Product Tape(s).
- An adequately defined input/output configuration data set (3081 Processor Complex users only). See "Considerations for Coding the Input/Output Configuration Source File" in Chapter 19 of the *VM/SP Planning Guide and Reference.*

Note: The scratch disks are located at real addresses 150, 151, and 152, respectively, when referenced in the Starter System examples (see Figure 3 on page 13 and Figure 4 on page 13).

Identify the Proper Starter System to Use

Starter systems that may be used with the installation procedure in "Chapter 2. Installation When Using the Starter System Tape" on page 15 are:

- 3310 Starter System
- 3370 Starter System
- 3330 Starter System
- 3340 Starter System
- 3350 Starter System
- 3375 Starter System

- 3380 Starter System

Note: Starter systems must be used with the DASD type for which they are designed. A starter system that was designed for a 3350 DASD cannot be used with a 3330 DASD.

From these Starter Systems you can generate a VM/SP system for residence on an FB-512, 3330, 3340, 3350, 3375, or 3380 volume. Because the system expects the term 'FB-512' in response to prompts, it is retained in this procedure (rather than changing to 'FBA'). An MSS 3330V volume cannot be used for system residence.

The sample procedure in Part 1 assumes that you are using a 3330 Release 3 Starter System. If not, you may have to adjust some suggested responses accordingly.

Each Starter System must be restored to a specific direct access device type. The 3330, 3340 and FB-512 Starter Systems are designed for the 3330 Model 1, 3340-70, and 3310 or 3370 devices. All work volumes are assumed to be of the same DASD type.

The installation procedure in this part assumes a merged VM/SP system Product Tape, containing the VM/SP CP and CMS components.

Note: This installation procedure is tailored to the uniprocessor user but it can be used by both the attached processor user and multiprocessor user.

Distribution Tapes Layouts

The Starter System is distributed on a 9-track tape (1600 or 6250 bpi) that can be restored to direct access volumes. You must have specified the device type feature code for FB-512, 3330, 3340, 3350, 3375, or 3380 when you ordered VM/SP.

1. The format of the Starter System Tape

   File 1: Stand-alone FORMAT/ALLOCATE program
   File 2: Stand-alone DDR program
   File 3: DDR backup of the Starter System

2. The format of the Product Tape

   File 1: Installation aids (PREP EXEC, PPPREP EXEC, INSTFPP EXEC)
   File 2: Sample files
   File 3: CP text files
   File 4: VM/SP HELP files
   File 5: CMS text, EXECs, macro libraries
   File 6: CP source (6250 bpi only).
   File 7: CMS source (6250 bpi only).

   Note: Source files are shipped on separate tapes if 1600 bpi required.

3. The system Program Update Tape (PUT)

   With your initial VM/SP shipment, you might receive a service tape(s) in PUT format. These tapes contain all source updates, text decks, modules, macros, macro libraries, and procedures required to build the latest levels of CP and CMS.

The layout of the Starter System's minidisk areas is determined by the number of cylinders that can be dumped onto one reel of 1600 bpi tape. Therefore, only selected cylinders are restored from the Starter System Tape. These areas (CP nucleus, 190 minidisk, etc.) are defined in VMUSERS DIRECT (this is the directory and is file 2 of the Product Tape), along with default locations for minidisks not restored using the DDR utility program.

Changes to Consider Before Starting

PREP EXEC automatically formats minidisks using the CMS FORMAT command, and loads the Product Tape to them using the VMFPLC2 command. If you have an existing VM/SP system, check the VMUSERS DIRECT file sample for compatibility with your system design.

The VMUSERS DIRECT file is shown in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

You can change the location and size of any minidisk before formatting and loading by using the System Product Editor and the DIRECT command.

Minidisks restored from the Starter System are:

190[1]: CMS minidisk containing all CMS files and the CMS nucleus.

191: Work disk for the MAINT userid.

Required minidisks allocated but not formatted are:

194: CP minidisk to contain all CP files required to build a CP nucleus.

19D: Default minidisk to contain HELP[2] files. Allocation for this minidisk is larger than required to allow for future growth, and for user-written and other selected product HELP files.

293[3]: CMS service minidisk.

294[3]: CP service minidisk.

Optional service minidisks are:

19E[4]: S-disk extension, used to contain user-selected products.

[1] The CMS nucleus area on the 190 minidisk has been expanded.

[2] VM/SP Release 1 used the 190 minidisk for HELP files.

[3] The reserved minidisk sizes may not be adequate for the duration of the release cycle and may need to be increased at some time.

[4] A sample allocation is illustrated in this procedure, but the actual size is dependent on the additional user-selected products to be installed onto the base system.

[5] VM/SP Release 2 used the 190 minidisk for EREP files.

201[5]: EREP library minidisk

29E: User-selected product service minidisk

39E: User-selected product source minidisk

393: CMS source minidisk

394: CP source minidisk

No paging space has been provided. Consideration for the allocation of paging space should be made and an entry into the SYSOWN list of DMKSYS should be made.

Before you attempt to generate a VM/SP system, make sure listings of the following files are available:

- Real I/O configuration file (DMKRIO ASSEMBLE)
- CP system control file (DMKSYS ASSEMBLE)
- VM/SP directory file (VMUSERS DIRECT)
- System name table file (DMKSNT ASSEMBLE)

The file listings are in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

A full screen editor, XEDIT, is available with the Starter System to assist in changing these files. Other files that you may wish to modify include the optional forms control buffer load file (DMKFCB) and/or the optional UCS/UCB/FOB load files (DMKUCS, DMKUCB, DMKUCC, DMKPIA, DMKPIB). You will be concerned about this only if you intend to change the Starter System defaults. Information about preparing these files is in "Part 2: Defining Your VM/SP System" of the *VM/SP Planning Guide and Reference.*

Meaning of First and Second Level Operation

First Level Operation Assumptions

When you run a system *first level,* it is assumed that you:

- Are sitting at the console terminal of your processor

- Have placed your processor in display mode

- Know how to operate the computer and all of its associated hardware devices.

This procedure does not provide any assistance for these tasks. If you are not sure of all basic functions, take the time to review them or have someone available to assist you with the operator functions such as tape and DASD mounting, system IPL and display of PSW's.

Second Level Operation Assumptions

When you run a system *second level*, you are running your system in a virtual machine under the Control Program (CP) of your current VM/SP system. In this environment, the computer operator attaches the required DASD and tape devices to your VM logon userid as required.

For example, three DASD devices and two tape drives could be attached to your virtual machine. The addresses for these devices are assigned to your virtual machine. Therefore, you can define these devices to match the examples provided in the installation procedure.

When running *second level*, it is important to keep in mind the level of CP on which you are running. The linend character is used in the installation procedure to pass commands to the specific level of CP you are addressing. That is why it is necessary to change the linend character of your *first level* CP to %. This leaves your *second level* CP with the linend character #.

Note: In this environment it is not advisable to do any formatting of CP. This would put an increased load on the system channels. Any CP formatting should be done during low production periods.

Relationship between First and Second Levels of Operation

<table>
<tr><td>STARTER SYSTEM
SECOND LEVEL CMS</td><td rowspan="2">FIRST LEVEL CMS
(Other Virtual Machines)</td></tr>
<tr><td>STARTER SYSTEM
SECOND LEVEL CONTROL PROGRAM
(TERM LINEND #)</td></tr>
<tr><td colspan="2">FIRST LEVEL CONTROL PROGRAM
(TERM LINEND %)</td></tr>
</table>

Meaning of the ATTN and ENTER Keys

Two functions common to both display and typewriter-like terminals are ATTENTION and ENTER (end-of-input) signaling. Pressing ENTER requests the system to accept the data typed on the command line. Pressing ATTENTION interrupts the system, so that you can do something else.

The keys that perform these two functions have different labels on different terminals, but to simplify things in this book:

ATTN refers to the attention-signaling key.

ENTER refers to the end-of-input signaling key.

The exact key(s) that perform these functions for your terminal or console are shown in the figure that follows.

Keys Used for Signaling

Terminal	ATTN	ENTER
1052	RESET LINE	RETURN or EOB
2741	ATTN	RETURN
3101	BREAK	BREAK, SEND, new-line
3210, 3215	REQUEST	END
3277	ENTER, DUP/PA1	ENTER
3278	ENTER, PA1	ENTER
3279	ENTER, PA1	ENTER
3767	ATTN	EOB, EOM
System/370 138, 148, 158	ENTER	ENTER

For this book the symbols ATTN and ENTER, respectively, are used to represent the terms "Press ATTN" and "Press ENTER" and related expressions. You should refer to this table when you are not sure which keys you need when requested for ATTN or ENTER.

Format of this Book

The procedures in this book are formatted in multiple columns to help you distinguish among:

- Messages
- Prompts for input
- Responses
- Comments related to specific actions
- General comments.

Messages and prompts are printed in uppercase lettering in the left column with a light blue background. Responses, also in the left column, are printed in dark blue lowercase type, indented, and followed by blank spacing. Responses often require two types of data entry:

- "literal" entries - You must enter the data exactly as is shown.

- "request" entries - You must supply and enter the data that pertains to your installation.

  Note: For the "request" entry indicated by ENTER, make a null entry by pressing your ENTER signaling key.

A sample of how each appears in this book follows:

```
PROMPT OR MESSAGE
```

literal entry response

request entry response

Information and comments relating to a specific message, prompt, or response are printed in the right column of the page. For example:

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
```

This message indicates that FORMAT/ALLOCATE program is now loaded and can be used.

Throughout the sample procedures presented in this book a set of *real* device addresses is used for consistency. You may substitute your own values for real device addresses given in the samples. For example:

cuu

cuu means that you may enter any legitimate device address that suits your installation's device addressing. The supplied virtual device addresses are addresses by which the Starter System accesses devices. They are shown in bold-faced type and must be entered as shown.

Descriptive information and comments not related to a specific message, prompt, or response (like this paragraph you are reading) are centered on the page.

To make the installation procedure easier to follow, some prompts and messages have been changed slightly from the actual displays. For example, message numbers and timestamps have been removed. Some trivial messages are not shown in this sample procedure. While the Starter System is processing, you may get these messages displayed. You can usually ignore them as noted in the procedure.

This procedure assumes:

- You are using a 3330 Starter System. If you use another type starter system, you might have to adjust your responses to some of the prompts in this procedure.

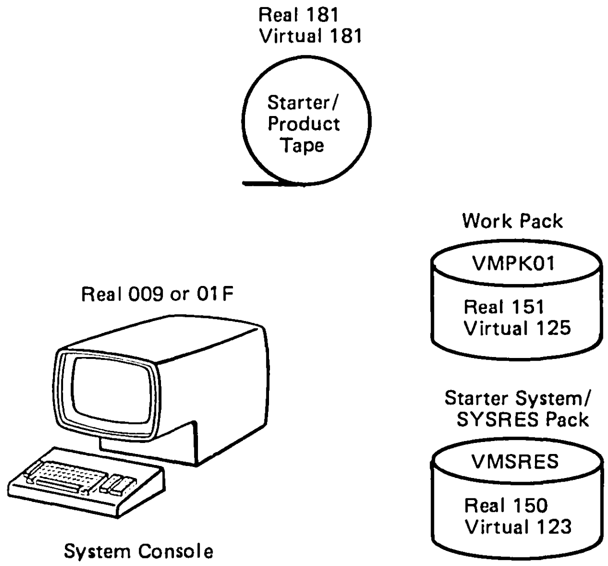
- You have selected the processor console display mode rather than the printer-keyboard mode IML (initial microprogram load) option.

- You are generating VM/SP from a real console at *first level* or you are generating VM/SP from a virtual machine at *second level.* For both levels of operation, some additional and possibly different commands may have to be entered at some points in a procedure. Also, some message contexts may vary from one level of operation to another.

Note: The stand-alone programs used in this procedure might not work on processors with the Loop Adapter feature enabled.

The steps in the sample system installation procedure are grouped into phases according to logical break points in the procedure. Flow charts at appropriate points in the procedure illustrate the phases and steps within each phase. This grouping should aid in determining a restart point in the event of a failure during the installation process.

Figure 3 on page 13 and Figure 4 on page 13 show the required and optional device configurations for generating VM/SP using a Starter System. The real address shown are those used in the sample procedures. Your installation may use different real addresses.



Real addresses shown are those used in the sample procedure.
Your installation may use other real addresses.

Figure 3. Required Devices for Generating VM/SP Using the Starter System

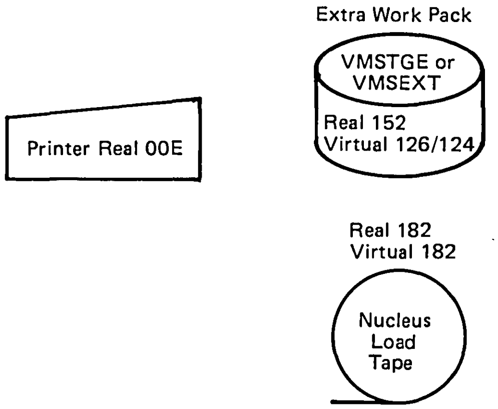


Figure 4. Optional Devices When Using a Starter System

Chapter 2. Installation When Using the Starter System Tape

Phase 1. Preinstallation Preparation

Before beginning the actual installation process, you should obtain the following:

- VM/SP Starter System Tape
- VM/SP Product Tape
- The number of recommended volumes, for your DASD type, to install VM/SP without any optional feature program products. See the table that follows:

DASD Type	Volumes if No Source Loaded	Volumes if Source Loaded
3310, 3340	3	5
3330	3	4
3350, 3370, 3375, 3380	2	3

- The recommended number of additional volumes, for your DASD type, to install your configuration of optional feature program products. See Appendix I, "Optional Feature Program Products" on page 287.
- One Tape Drive (or more if possible)
- One display terminal from which to enter the necessary commands
- A copy of the chart in Appendix H, "Installation Reference Worksheet" on page 285. This volume number, address, label, and device type information will be needed during the installation of VM/SP.

Review the default DMKxxx files for your DASD type to decide if you can use the system defaults. The DMKxxx files are on the MAINT 191 minidisk. The DMKxxx file listings are available in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

Before choosing to use the defaults, you must be sure that the I/O configuration is suitable for your installation. If it is, you can take the system defaults when you install. If not, you need to create or modify your own DMKRIO. See "Chapter 19. Preparing the Real I/O Configuration File (DMKRIO)" in the *VM/SP Planning Guide and Reference* for more information.

Use the following criteria to help you decide whether the defaults are to be used.

- Your real DASD addresses must match the addresses given in the default DMKRIO.

- Your real operator's console must match either the console or an alternate console defined in the RIOGEN macro of DMKRIO.

- You must have TAPES and GRAPHICS at addresses defined in the DMKRIO.

If these conditions are met, you will be able to use the default DMKRIO to build your system. Tailoring of DMKRIO can be done after the system is running if you want to add or delete devices. If you do not want the defaults or the criteria was not met, you may create a DMKRIO by altering the default file using the System Product Editor.

If you are going to create a new DMKRIO file, you must know your system and its features very well. This includes all of the devices, addresses, and control units.

In the procedure that follows, you will be required to provide addresses and device types for the new system you are building. If you do not plan to use the available system defaults and/or if you prefer to label or address your devices and volumes differently, record the changes you are planning before you begin. Appendix H, "Installation Reference Worksheet" on page 285 is provided for your convenience so that you will have a record of your plans at the start of the procedure.

Phase 2. Install the Starter System

Phase 2 Overview

In Phase 2 you will be doing:

Step 1. Load and Run the Format Program

Step 2. Restore the Starter System to Disk

Step 3. Load the CP Nucleus

Step 4. Load the CMS Nucleus

The instructions for completing these steps are on the pages that follow.

To improve readability, some prompts and messages in this procedure are modified slightly from the actual displays. For example, message numbers and timestamps have been removed.

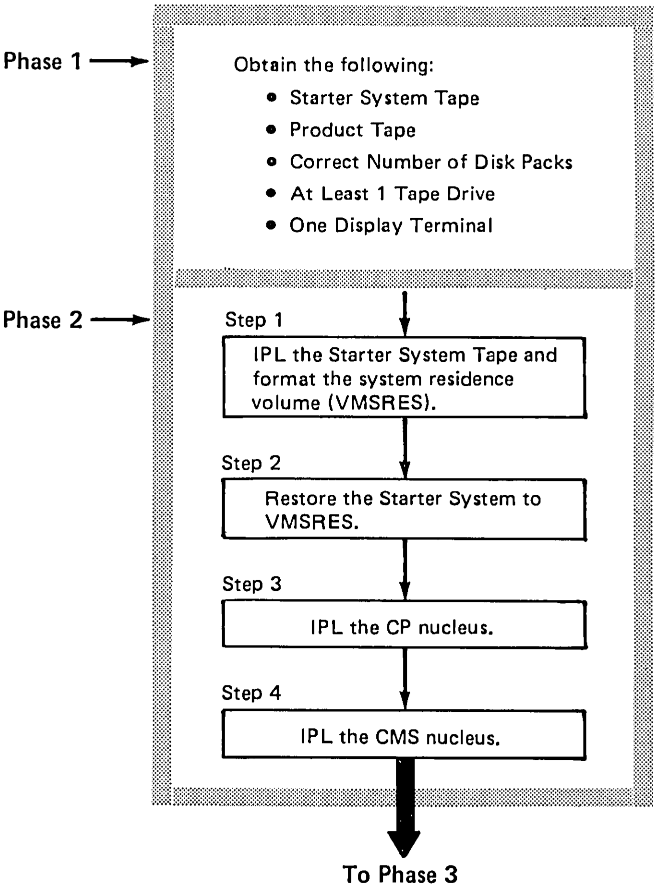


Figure 5. Format VMSRES and IPL Starter System

Step 1. Load and Run the Format Program

If your console is not addressed 009 or 01F, remember to press the REQUEST key or its equivalent before proceeding. This is necessary to identify it as the system console.

Figure 6 shows the Starter System Tape format. The FORMAT/ALLOCATE program is the first file on the tape. *Do not rewind the tape after completing this step.*

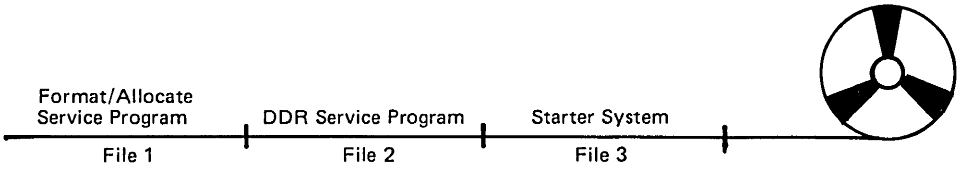


Figure 6. VM/SP Starter System Tape Format

Format and Label the Starter System/System Residence Volume (VMSRES)

The Starter System, when restored, will be the base of the new system residence volume. In this procedure the system residence volume is called VMSRES. VMSRES is formatted in this portion of Step 1.

You will use the FORMAT/ALLOCATE program to format and label the new system residence volume. To execute the FORMAT/ALLOCATE program, respond to the prompting messages at your console.

Note: Formatting large DASD devices can take up to 30 minutes. During this time, the SYSTEM light is on, but there is no screen activity to inform you about the progress of the formatting process. When formatting is done, you will get a message. If, during the formatting process, a problem occurs and you cannot proceed, rewind the Starter System Tape and start over at the beginning of Step 1.

Load the FORMAT/ALLOCATE program from the Starter System Tape in the following manner:

1a. Mount and ready the DASD volumes that are to be used.

1b. Mount and ready the Starter System Tape.

1c. IPL (Initial Program Load) the tape.

- At *first level*, IPL from a tape device according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)
- At *second level*, define VMSRES at virtual address 123 if not previously done and attach a tape drive to your virtual machine.

def cuu 123	cuu is the present virtual address of your VMSRES volume.

Then, mount and IPL the Starter System Tape.

ipl cuu

cuu is the address of the tape drive attached to your virtual machine.

Note: If nothing seems to be happening after IPL, the system is most likely in a wait state. A wait state exists if:

- The SYSTEM light has turned off.
- The WAIT light has turned on.
- The system appears to be inactive.

To initiate the FORMAT/ALLOCATE program from a wait state:

- At *first level*: ENTER
- At *second level*: ATTN (For example, press PA1 on 3278 terminal). Then, ENTER

All standalone programs have limited error recovery routines. If your system fails to respond after the wait state is entered, it could also be a media or tape error. If so, re-IPL the tape.

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:
```

This message indicates that FORMAT/ALLOCATE program is now loaded and can be used.

format

```
FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real disk address on which the new system will be built (VMSRES).

```
ENTER DEVICE TYPE:
```

dt

dt is the device type of the DASD to be formatted. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512. For 3310 or 3370 devices you must enter 'FB-512' when prompted for device type.

```
ENTER START CYLINDER (XXX) OR ''LABEL'':
```

ENTER

The system will display: `ENTER START PAGE (XXX) OR "LABEL":` for FB-512 devices (3310 or 3370). The null entry defaults to 000.

```
ENTER END CYLINDER (XXX):
```

ENTER

The system will display: `ENTER END PAGE (XXX)` for FB-512 devices (3310 or 3370). Based on the device type you previously entered, the null entry results in one of the following values:

(CKD devices)		**(FB-512 devices)**	
3330	end cyl. 403	3310	end page 15751
3330-11	end cyl. 807	3370	end page 69749
3340-35	end cyl. 347		
3340-70	end cyl. 695		
3350	end cyl. 554		
3375	end cyl. 958		
3380	end cyl. 884		

Step 1

```
ENTER DEVICE LABEL:
```

vmsres

```
FORMAT STARTED

FORMAT DONE
000 NO. PAGE RECORDS WITH READ-CHECK
ERRORS
```

If the previous message indicates other than **000 NO. PAGE RECORDS WITH READ-CHECK ERRORS**, discontinue the installation process and contact your Systems Engineer or hardware service personnel. (You may have to initialize the volume with the Device Support Facilities program. Refer to the *Device Support Facility User's Guide and Reference.*)

Note: Formatting large disks can take up to 30 minutes.

At completion of Step 1 your console displays:

```
ENTER FORMAT OR ALLOCATE:
```

At *first level*, proceed to Step 2.

At *second level* press **ATTN** then **ENTER**

Note: You may want to format additional volumes at this time while the format/allocate program is still operational.

Proceed to Step 2.

Step 2. Restore the Starter System to Disk

In this procedure, volume uses are as follows:

VMSRES

Holds the contents of the starter system tape and the product tape until the starter system portion of the installation procedure completes. The starter system then becomes the operational system.

VMPK01

Required for installation using the starter system. The PREP EXEC and GENERATE EXEC use TEMP and T-DISK space allocated on this pack for work space.

VMSEXT

Used for additional work space, or to hold program products if a 3310, 3330, or 3340 is used as the SYSRES device. This pack is used but not required for the installation procedure.

VMSTGE

Used to hold source code. If your installation does not require source code to be maintained on DASD, this pack is not required.

Warning: The VMPK01 and VMSEXT volids are contained in the starter system SYSOWN list. Therefore, these work volumes must not be labeled VMPK01 or VMSEXT if they are to be formatted. If they are, use the FORMAT/ALLOCATE program to relabel these volumes prior to proceeding, or disable these volumes prior to proceeding with step 3.

The following are sample allocations for VMSRES:

	3310*	3330	3340-70	3350	3370*	3375	3380
perm	00002 11559	000 199	000 069	000 100	00002 11559	000 143	000 096
temp	--- ---	200 274	--- ---	--- ---	--- ---	--- ---	--- ---
drct	11560 11787	275 278	070 079	101 102	11560 11787	144 147	097 098
tdsk	--- ---	--- ---	--- ---	--- ---	--- ---	148 287	--- ---
perm	11788 14464	279-403[1]	080 094	103 233	11788 11851	--- ---	099 401
temp	14465 15274	--- ---	095 124	234 276	11852 15274	288 478	402 441
perm	15275 15751	279 807[2]	125 695	277 277	15275 15751	479 480	442 442
temp				278 317	15752 34501	481 555	443 482
perm				--- ---	34502 47540	556 958	--- ---
tdsk				318 417	47541 51915		483 582
perm				418 554	51916 69749		583 884
end							

* 3310 and 3370 are FB-512 devices, all others are CKD devices.
The allocations for FB-512 devices are expressed as blocks.
The allocations for CKD devices are expressed as cylinders

[1] 3330 Model 1

[2] 3330 Model 11

The following are sample allocations for VMPK01:

	3310*	3330	3340-70	3350	3370*	3375	3380
perm	00002 03421	000 119	000 285	000 226	00002 14458	000 226	000 326
tdsk	--- ---	120 214	--- ---	--- ---	14459 26958	--- ---	--- ---
temp	03422 05921	215 334	286 366	227 326	26959 36958	227 326	327 426
tdsk	--- ---	--- ---	367 616	327 554	--- ---	327 958	427 884
perm	05922 11374	335 403[1]	617 695		36959 69749		
tdsk	11375 15751	--- ---	--- ---		--- ---		
perm		335 807[2]					
end							

* 3310 and 3370 are FB-512 devices, all others are CKD devices.
The allocations for FB-512 devices are expressed as blocks.
The allocations for CKD devices are expressed as cylinders

[1] 3330 Model 1

[2] 3330 Model 11

VMSTGE and/or VMSEXT may be formatted and allocated by using the Format/Allocate program with the data from the appropriate table below.

The following are sample allocations for VMSEXT:

	3310*	**3330**	**3340-70**	**3350**	**3370***	**3375**	**3380**
perm	00002 10030	000 000	000 464	--- ---	--- ---	--- ---	--- ---
tdsk	--- ---	001 110	465 625				
perm	--- ---	111 296	626 695				
temp	10031 12530	297 391					
perm	12531 15751	392 403[1]					
perm		392 807[2]					
end							

* 3310 and 3370 are FB-512 devices, all others are CKD devices.
The allocations for FB-512 devices are expressed as blocks.
The allocations for CKD devices are expressed as cylinders

[1] 3330 Model 1

[2] 3330 Model 11

The following are sample allocations for VMSTGE:

	3310*	**3330**	**3340-70**	**3350**	**3370***	**3375**	**3380**
perm	00002 15751	000 403[1] 000 807[2]	000 695	000 554	00002 69749	000 958	000 884

* 3310 and 3370 are FB-512 devices, all others are CKD devices.
The allocations for FB-512 devices are expressed as blocks.
The allocations for CKD devices are expressed as cylinders

[1] 3330 Model 1

[2] 3330 Model 11

Step 2

Load the DASD DUMP RESTORE program (second tape file) from the Starter System Tape.

- If you are running *first level*, IPL your tape drive according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)
- If you are running *second level*, enter:

  ipl cuu

  cuu is the address of the tape drive attached to your virtual machine.

Note: You may have to IPL the tape drive twice to get past the tape mark. Also, you may need to press the ENTER key when the system goes into a wait state in order to initiate the DASD DUMP/RESTORE program.

Restore the VMSRES volume by responding as follows to the DASD DUMP/RESTORE program prompts:

```
VM/370 DASD DUMP/RESTORE PROGRAM - VM/SP
ENTER CARD READER ADDRESS OR CONTROL STATEMENTS
ENTER:
```

sysprint cons

```
ENTER:
```

input cuu nnnn

cuu is the address of the tape drive containing the starter system tape.
nnnn is the type of the tape drive: 2400, 2420, 3420, 3430, or 8809. (Specify 3410 as 3420.)

```
ENTER:
```

output cuu dt vmsres

cuu is the address of the disk drive on which the Starter System is to be restored.
dt is the device type of the DASD to be restored. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512 for 3310 or 3370.

```
ENTER:
```

restore all

```
RESTORING VMSRES
DATA DUMPED MM/DD/YY AT HH.MM.SS GMT FROM VMSRES RESTORED TO VMSRES
INPUT CYLINDER EXTENTS      OUTPUT CYLINDER EXTENTS
        START     STOP       START     STOP
        ....      ....       ....      ....
        ....      ....       ....      ....
        ....      ....       ....      ....
END OF RESTORE
```

The start and stop cylinder/block extents will vary, depending on the Starter System device type. Also, the block extents may not be displayed if the console is in DISPLAY mode.

ENTER:

ENTER

```
END OF JOB
```

If you need to restart the Starter System restore process, do the following:

1. Ready the Starter System Tape.
2. IPL the Starter System tape drive to bypass the FORMAT/ALLOCATE program.
3. Start over at the beginning of Step 2 on page 21.

Otherwise, proceed to the next page.

Step 2

If you are restoring to a 3310, read this page; otherwise, skip this page.

3310 Reallocation

If you are restoring a 3310 Starter System, complete this section. *Otherwise*, proceed to the next page.

- Rewind the Starter System Tape.
- IPL the Starter System Tape.

Reply to the following prompts:

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:
```

allocate

```
ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real disk address on which the Starter System was just restored.

```
ENTER DEVICE TYPE:
```

fb-512

```
ENTER DEVICE LABEL:
```

vmsres

```
ENTER ALLOCATION DATA FOR VOLUME VMSRES
TYPE PAGE PAGE
```

Forces allocation table to be rewritten.

perm 2 11559

```
ENTER:
```

end

```
ALLOCATION RESULTS
PERM        XXXXX    XXXXX
DRCT        XXXXX    XXXXX
PERM        XXXXX    XXXXX
TEMP        XXXXX    XXXXX
PERM        XXXXX    XXXXX
DEVICE CUU VOLUME VMSRES   ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Continue with Step 3 on page 28 to IPL the Starter System and define the devices required for the system installation.

If you have a 3330-11 device type, read this page; otherwise, go to Step 3.

3330-11 Reallocation

If you are restoring a 3330-11 Starter System, complete this section. *Otherwise*, proceed to the next page.

The portion of the disk space from cylinder 404 to cylinder 807 on the VMSRES volume should be re-allocated for minidisk use.

- Rewind the Starter System Tape.
- IPL the Starter System Tape.

Reply to the following prompts:

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:

  allocate

ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):

  cuu
```

cuu is the real disk address on which the Starter System was just restored.

```
ENTER DEVICE TYPE:

  3330-11

ENTER DEVICE LABEL:

  vmsres

ENTER ALLOCATION DATA FOR VOLUME VMSRES
TYPE CYL CYL
.... ... ...

 perm 404 807
```

Then press ENTER.

```
 end

ALLOCATION RESULTS
PERM        XXXXX   XXXXX
DRCT        XXXXX   XXXXX
PERM        XXXXX   XXXXX
TEMP        XXXXX   XXXXX
PERM        XXXXX   XXXXX
DEVICE CUU VOLUME VMSRES   ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Continue with Step 3 on the next page to IPL the Starter System and define the devices required for the system installation.

Step 3. Load the CP Nucleus

General Considerations Before IPLing the Starter System

If you have control units that share more than 16 devices, and are also switchable to a second processor (non-IPL), care needs to be taken. While you perform the system generation, put the channel-interface-enable switch on the non-IPL processor in the DISABLE position. Any other loosely-coupled processor (via channels) should be placed in STOP mode during actual IPL of the system.

Note: An IPL in a second level environment takes longer than you might expect. The symptom appears as a performance degradation or hung condition. It is caused by contention for service by the devices on the shared control units. Also, if the missing interrupt handler (MIH) is turned on, message MSDMKDID546I (INTERRUPTION CLEARED) appears repeatedly. This is not an indication of a problem; the system is operating correctly.

Warning: If you plan to execute PREP CPFMT, disable all volumes already labeled *VMSEXT and/or VMPK01*. To do this:

- **At *first level*, set the appropriate channel ENABLE/DISABLE switch or switches on your control unit to the DISABLE position.**
- **At *second level*, define the volumes in Step 3 at virtual addresses other than those you chose in Steps 1 and 2. Later in the process, these volumes will be redefined and attached as required.**

If you are operating at *first level*, proceed to page 31 to IPL and define your starter system. If at *second level*, continue below.

Considerations Regarding Second Level Operation

The formatting of volumes and minidisks with *second level* operations causes overhead at all levels. Expect extremely long job completion times. It is recommended that these steps *not* be done at *second level*, if possible.

If, however, you choose to operate at *second level*, some additional entries are suggested depending on whether you are operating with local or remote terminals.

- **Running a Second Level Operation with Local Terminals**

To have full screen System Product Editor support on local devices when operating at *second level*, complete the following procedure.

Note: All devices from Figure 3 on page 13 and Figure 4 on page 13 must be attached to the userid *after* logon.

logon userid	Any valid directory entry can be used if you are not currently logged on to your *first level* system.
set ecmode on	
def 01f 009	

Warning: The graphics device (secondary console) must not be addressed with the same first two prefix symbols used in the address of the primary console device. For example, the console device at address 01F disallows any graphics address of the form 01x. Thus, all graphics device addresses ranging from 010 to 01F are not usable for console address 01F.

To bypass this problem, you must either redefine the primary console device address to 009 as shown above or use the default address of 020 to define the graphics (secondary) console.

Check your primary console address by issuing:

query console

Use the following table to decide which address to use for your secondary console.

If primary console is	**Then secondary console is**
01F	**020 or greater**
009	**010 or greater**

For more detail, refer to the cautionary notes under "DEDICATE Control Statement" (Chapter 18) and under "Miscellaneous Restrictions" (Appendix D) in the *VM/SP Planning Guide and Reference.*

Continue by entering the next command.

term conmode 3270[6]	Display unit must be local non-SNA device.
term linend %	Allows the # sign to be used for *second level* linend.
define stor 1m	Minimum; can be larger if needed.

[6] If the message **DMKCFT006E INVALID DEVICE TYPE - nnnn** appears, continue in the procedure for "Remote Terminals" on the next page (just prior to "term linend %").

- **Running a Second Level Operation with Remote Terminals**

To have full-screen support when executing the starter system from a virtual machine on an existing system using remote terminals, complete the procedure below.

Note: All devices from Figure 3 on page 13 and Figure 4 on page 13 must be attached to the userid *after* logon.

Enter these commands from the primary user console:

```
logon userid              Use any valid directory entry.

set ecmode on
```

This is the entry point for users who had the **INVALID DEVICE TYPE - nnnn** message from the procedure for "Local Terminals."

```
term linend %             Allows the # sign to be used for second level CP
                          linend.

def graf 020
```

IPL the starter system (VMSRES). This causes the starter CP nucleus to load.

```
ipl cuu                   cuu is the virtual address of VMSRES.

vary on 020

enable 020
```

Once the system is up, log off the current userid MAINT on the master console.

```
dial userid               Same as logon userid above.

logon maint cpcms         Logging on another physical console as your graphics
                          device using your MAINT userid.
```

This allows full console support on the second device. All running will be done on this device, but the primary console must remain active.

IPL and Define Your Starter System

* * * * RESTART HERE * * * *

This is the restart point if you have a failure while building the Starter System.

IPL the address of the VMSRES Starter System volume restored in the previous step.

1. If you are running *first level*, IPL your disk drive according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)

2. If you are running *second level*, ATTN. Then enter:

   ipl cuu clear Causes the starter CP nucleus to be loaded. cuu is the real address for VMSRES.

If no messages appear on the operator's console, and the system goes into a wait state (PSW X'27'), press the ENTER key at the operator's console. When the Starter System has been IPLed, the following prompt will be issued:

```
VM/SP: STARTER SYSTEM
```

If the system does not IPL, verify that the real console address matches the console address in DMKRIO.

If this is your first IPL of the Starter System Tape, the procedure continues at the top of the next page. The prompt shown below occurs *only* on the **second** and subsequent IPLs of the Starter System.

```
*** DO YOU WISH TO RE-DEFINE YOUR SYSTEM *** (YES|NO):
```

yes **yes** performs the steps starting on the next page.
no bypasses them and puts you at the top of page 34.

Note: To use the Starter System defaults for device type and address prompts, **ENTER** for each prompt.

If you do not have a printer, a card reader, punch, or second tape drive, simply **ENTER** to generate the default device type. These devices are not required to generate the system.

```
ENTER PRINTER ADDRESS (CUU):
```

cuu

cuu is the address of your printer. (00E is the default.)

```
ENTER DEVICE TYPE (1403,1443,3203,3211,3262,3289E,3800,4245,4250):
```

xxxx

xxxx is the printer device type. (1403 is the default.)

```
ENTER PUNCH ADDRESS (CUU):
```

cuu

cuu is the address of your card punch. (00D is the default.)

```
ENTER DEVICE TYPE (2540P,3525):
```

xxxx

xxxx is the punch device type. (2540P is the default.)

```
ENTER READER ADDRESS (CUU):
```

cuu

cuu is the address of your card reader. (00C is the default.)

```
ENTER DEVICE TYPE (2540R,2501,3505):
```

xxxx

xxxx is the reader device type. (2540R is the default.)

```
ENTER ADDRESS WHERE FIRST TAPE IS MOUNTED (CUU):
```

cuu

cuu is the address of your Starter System Tape. (181 is the default.)

```
ENTER DEVICE TYPE (3420,2415,2420,2401,3430,8809:)
```

xxxx

xxxx is the device type of the tape drive. (3420 is the default.)

```
ENTER ADDRESS OF A SECOND TAPE DRIVE (CUU):
```

cuu

cuu is the address of another tape drive. Do NOT specify the same address used in the previous question. (182 is the default.)

```
ENTER DEVICE TYPE (3420,2415,2420,2401,3430,8809:)
```

xxxx

xxxx is the device type of the tape drive. (3420 is the default.)

```
ENTER DEVICE ADDRESS OF WORK PACK (CUU):
```

cuu

cuu is the address of VMPK01 for 3350, 3370, 3375 or 3380.
cuu is the address of VMSEXT for 3310, 3330 or 3340.
(151 is the default.)

```
ENTER DEVICE TYPE (3330, 3340, 3350, 3375, 3380, 2305, FB-512):
```

xxxx

xxxx is the DASD device type.
(3330 is the default.)[7]

```
ENTER ADDRESS WHERE EXTRA WORK PACK IS MOUNTED (CUU):
```

cuu

cuu is the address of VMPK01 if 3310, 3330 or 3340. ENTER for cuu if 3350, 3370, 3375 or 3380.
(152 is the default.)

```
ENTER DEVICE TYPE (3330, 3340, 3350, 3375, 3380, 2305, FB-512):
```

xxxx

xxxx is the DASD device type.
(3330 is the default.)[7]

Warning: The first two prefix symbols used in the address of the graphic device and the primary console cannot be the same. A violation of this restriction may cause your system to ABEND. See page 29 if you need help deciding which graphics device address to use.

```
ENTER ADDRESS OF A GRAPHIC DEVICE (CUU):
```

cuu [8]

cuu is the address of a graphic device other than the system console.
(020 is the default.)

```
ENTER DEVICE TYPE (3277, 3278, 3066):
```

xxxx

xxxx is the graphic device type.
(3277 is the default.)

```
***SYSTEM DEFINITION COMPLETED***

CUU PRINTER
CUU PUNCH
CUU READER
CUU FIRST TAPE
CUU SECOND TAPE
CUU WORK PACK
CUU EXTRA WORK PACK
CUU GRAPHIC DEVICE
ARE THE ABOVE ENTRIES CORRECT (YES,NO):
```

yes (or **no**)

If you respond **no**, you will be restarting in Step 3 on page 31 (See RESTART HERE).

[7] Depending on the starter system, a reply of 3330 may return a prompt for the model type.

[8] If the graphics device to be defined is on the same subchannel as your operator's console on your *second level* system, you will abend the system. To avoid this, define a graphics device address that is not on the same control unit as your *second level* operator's console address. See page 29 if you need help deciding which graphics device address to use.

```
VM/SP RELEASE X SERVICE LEVEL X XXX; MM/DD/YY HH:MM:SS

NOW HH:MM:SS EDT DAYOFWEEK MM/DD/YY
CHANGE TOD CLOCK (YES|NO) :
```

yes (or **no**)

At *first level*, to change the TOD clock respond **yes,** and enter the local date and time in response to **SET DATE** and **SET TIME** messages. Check the operation instructions for the exact method required for your installation.

At *second level*, always answer **no** (or **ENTER**).

```
DMKCPI971I SYSTEM IS UP GENERATED
HH:MM:SS START ((COLD|WARM|CKPT|FORCE) (DRAIN))|(SHUTDOWN) :
```

cold

Note: You can ignore any DMKLNKXXXE messages shown.

```
HH:MM:SS AUTO LOGON   ***   MAINT   USERS = 001  BY  SYSTEM
HH:MM:SS
DMKCPI951I CP VOLID VMSEXT NOT MOUNTED
DMKCPI951I CP VOLID VMPK01 NOT MOUNTED
```

Even though your processor may have *more* than 2048K real storage, the following message will appear:

```
DMKCPI952I 2048K SYSTEM STORAGE

DMKCPI957I STOR SSSK, NUC NNNK, DYN DDDK,
           TRA TTTK,FREE FFFK, V=R VVVK
HH:MM:SS FILES:  NO RDR,  NO PRT,  NO PUN
```

The following message will be displayed to indicate that the error recording area is being formatted.

```
HH:MM:SS FORMATTING ERROR RECORDING AREA
DMKCPI966I INITIALIZATION COMPLETE
```

If you have a failure at this time, recover by reIPLing the Starter System.

The following message will appear when you reIPL:

```
*** DO YOU WISH TO RE-DEFINE YOUR SYSTEM *** (YES|NO):
```

Respond **yes** if the device addresses are not specified correctly, or if the tapes and/or disks have been moved. If you reply **yes**, you will be restarting in Step 3 on page 31 (See RESTART HERE). Otherwise, reply **no**.

Note:

If you previously disabled the volumes labeled VMSEXT and/or VMPK01, and wish to use them as 'work' volumes, enable them now. If you didn't disable the volumes labeled VMSEXT and/or VMPK01, continue with Step 4 on the next page.

To enable the volumes:

- At *first level*, set the appropriate channel ENABLE/DISABLE switch or switches on your control unit to the ENABLE position.

- At *second level*, attach the volumes redefining the volume addresses to those you entered in reply to the Starter System messages on the previous pages. Then, vary on the *second level* system by entering the following:

def cuua cuu1	cuua is the work volume address used in Step 3. cuu1 is your work volume address.
def cuub cuu2	cuub is the extra work volume address used in Step 3. cuu2 is your extra work volume address.

vary on cuu1

vary on cuu2

att cuu1 * cuu1

att cuu2 * cuu2

Continue with Step 4 on the next page.

Step 4. Load the CMS Nucleus

You are now ready to load CMS. The Starter System is running and you are logged on as user MAINT automatically.

At this time your console displays:

```
VM/SP n STARTER

   ENTER

R;
```

The CMS nucleus is now loaded.

Continue with Phase 3 on the next page.

Phase 3. Install the Product Tape

Phase Overview

In Phase 3 you will be doing:

Step 5. Load the PREP EXEC.

Step 6. Format DASD and load the product tape.

The instructions for completing these steps are on the pages that follow.

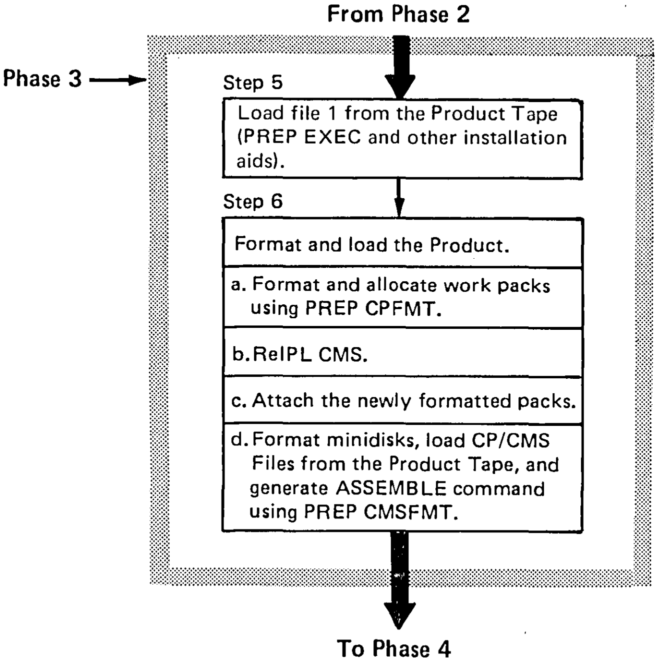

From Phase 2
Phase 3
Step 5
Load file 1 from the Product Tape (PREP EXEC and other installation aids).
Step 6
Format and load the Product.
a. Format and allocate work packs using PREP CPFMT.
b. ReIPL CMS.
c. Attach the newly formatted packs.
d. Format minidisks, load CP/CMS Files from the Product Tape, and generate ASSEMBLE command using PREP CMSFMT.
To Phase 4

Figure 7. Install Product Tape

Warning: The PREP EXEC changes your reader, printer and punch to Class I. In some instances the class might not be reset. If later work requires a class other than Class I, reset the class after executing PREP. For example, issue CHANGE PRINTER CLASS c1 where c1 is the selected spool class. Similar commands for your reader and punch also should be issued.

Step 5. Load PREP EXEC and Other Installation Aids

Issue the commands:

sp con start

This command allows you to save a copy of the console activity.

term mode vm

This command allows CMS commands to be issued successfully.

Note: If your console is in Printer-Keyboard mode (a processor IML Option), press the REQUEST key before entering each command.

*You are now finished with the Starter System Tape and will need to mount the Product Tape.*If another tape drive is available, you can mount the Product Tape (without ring) at the second address that you specified in Step 3. If only one tape drive is available to your system, remove the Starter System Tape and mount and ready the Product Tape on the same drive.

The Product Tape format is shown in Figure 8.

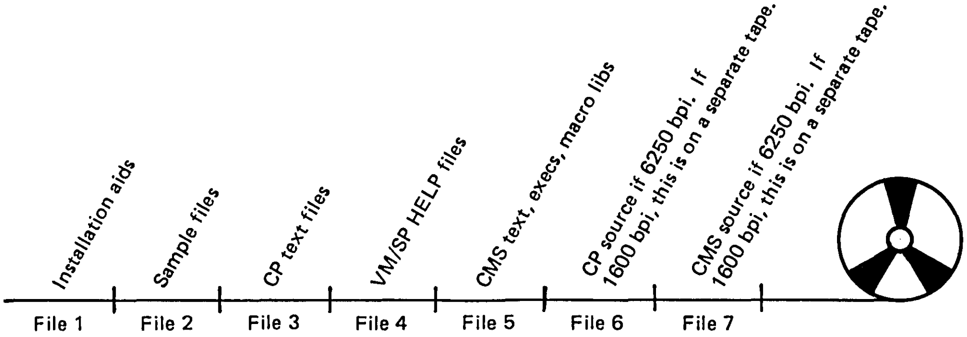

Installation aids
Sample files
CP text files
VM/SP HELP files
CMS text, execs, macro libs
CP source if 6250 bpi. If 1600 bpi, this is on a separate tape.
CMS source if 6250 bpi. If 1600 bpi, this is on a separate tape.
File 1
File 2
File 3
File 4
File 5
File 6
File 7

Figure 8. VM/SP Product Tape Format

Enter the following to continue:

vary on cuu

cuu is an address of a tape drive that you entered in Step 3. It should contain your newly mounted Product tape.

```
HH:MM:SS CUU VARIED ONLINE
```

attach cuu * 181

cuu is the address of the tape drive that you just "VARIED ON" in the previous response.

```
HH:MM:SS TAPE CUU ATTACH TO MAINT   181
```

Load the first tape file from the Merged Product Tape onto MAINT's 191 'A' disk. To do this, issue the command:

vmfplc2 load

```
LOADING.....
PREP      EXEC      A2
PPPREP    EXEC      A0
INSTFPP   EXEC      A1
PROFILE   EXEC      A1
FEATURE$ PRODUCTS A1
CMSUSER   PROFILE  A1
X$INST$X XEDIT     A1
END-OF-FILE OR END-OF-TAPE
R;
```

The **R;** message indicates that the tape file is loaded and ready to use.

Note: From this point on, you will receive error messages about MAINT's 319 and 325 disks whenever you IPL CMS. You may ignore these messages; you will no longer receive them after the disks are formatted later.

Continue with Step 6.

Step 6. Format DASD and Load the Product Tape

Note: Much of the manual entry of CMS commands required for installation in Release 1, has been eliminated by PREP EXEC. PREP EXEC automates those steps needed to generate VM/SP. For further insight into exactly what the PREP EXEC does and the CMS commands it issues, print a copy of it and follow the generation procedure in detail.

The PREP EXEC options for generating a VM/SP system using the Starter System Tape are:

A. PREP CPFMT - Formats and allocates work volumes.
B. PREP CMSFMT - Loads sample files to 191 minidisk and/or formats minidisks for receiving combinations of CP and CMS files.
C. PREP LOAD - Loads the Product Tape onto minidisks.

To initiate any PREP EXEC task, enter the PREP command with the appropriate operand, for example, PREP LOAD. If you need more information about a specific command, you can enter the HELP option following the operand.

For example, if you want to know more about PREP LOAD, you enter PREP LOAD HELP. Enter PREP HELP for further PREP function descriptions and options.

Prompts guide you through execution of each of these options. Some prompts and messages in this procedure are changed slightly from the actual displays in order to make them easier to read. For example, message numbers and timestamps have been removed.

A. PREP CPFMT - Formats and Allocates Work Volumes

Note: If a failure occurs during formatting and allocating of the work volumes, the PREP CPFMT command can be safely executed if you first detach MAINT 124 and MAINT 125.

To begin execution of the first PREP option issue:

prep cpfmt

```
NO RDR FILES
CAUTION: THIS EXEC PURGES READER FILES.
CONTINUE? (YES | NO)
```

yes

If Class I reader files do exist, you may wish to respond **no** to the **CONTINUE?** prompt. If so, load the files to prevent them from being erased. Then, continue the procedure by reissuing **prep cpfmt**.

```
' CPFMT ' PROCESS STARTING ...
DASD TYPE = 3330
ENTER THE ADDRESS OF THE VOLUME
THAT IS TO BECOME 'VMSEXT' (CUU)
OR 'SKIP' IF SOURCE CODE WILL
BE LOADED DURING THE
INSTALLATION PROCEDURE.
```

cuu

The VMSEXT message appears for 3310, 3330, or 3340 Starter Systems only. A **skip** response to this prompt results in a prompt, later in this step, for the address of VMSTGE (alternate for VMSEXT).

cuu is the device address of your work volume.

```
CUU VARIED ONLINE
DASD CUU ATTACH TO MAINT 124

ENTER THE ADDRESS OF THE VOLUME
THAT IS TO BECOME 'VMPK01' (CUU)
OR 'SKIP' TO BYPASS FORMATTING OF THIS
PACK.  THIS PACK IS REQUIRED
FOR STARTER SYSTEM WORK SPACE.
```

cuu

cuu is the device address of your extra work volume.

```
CUU VARIED ONLINE
DASD CUU ATTACH TO MAINT 125
IS DEVICE 'CUU' A MODEL 1 OR
MODEL 11 3330 ? (1 or 11)
```

This question displays for 3330 devices only.

11

```
IS DEVICE 'CUU' A MODEL 1 OR MODEL 11 3330 ? (1 or 11)
```

11

```
NO FILES PURGED
PUN FILE nnnn TO MAINT COPY 001 NOHOLD
```

From this point, formatting and allocation of work disks through the execution of PREP CPFMT is fully automated. Automated responses are supplied to the remaining PREP CPFMT prompts. These responses are shown just as displayed on your console.

Note: This is another reminder that formatting may take up to 30 minutes per volume. During that time no status messages are issued.

```
VM/370 FORMAT/ALLOCATE PROGRAM
ENTER FORMAT OR ALLOCATE:
```

No reply necessary to this response.

```
FORMAT
```

Begin formatting VMSEXT. For a 3370 Starter System, additional VMSRES allocation messages appear at this time.

```
FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
124
ENTER DEVICE TYPE:
3330-11
ENTER START CYLINDER (XXX) OR "LABEL":
000
ENTER END CYLINDER (XXX):
807
ENTER DEVICE LABEL:
VMSEXT
FORMAT STARTED

FORMAT DONE
```

VMSEXT formatting complete.

```
000 NO. PAGE RECORDS WITH READ-CHECK ERRORS
ENTER FORMAT OR ALLOCATE:
                                           No reply necessary to this response.

ALLOCATE                                   Begin allocating VMSEXT.

ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
124
ENTER DEVICE TYPE:
3330-11
ENTER DEVICE LABEL:
VMSEXT
ENTER ALLOCATION DATA FOR VOLUME VMSEXT
TYPE CYL CYL
.... ... ...
PERM 000 000
TDSK 001 110
PERM 111 296
TEMP 297 391
PERM 392 807
END
ALLOCATION RESULTS
PERM 000 000
TDSK 001 110
PERM 111 296
TEMP 297 391
PERM 392 807

DEVICE 124 VOLUME VMSEXT ALLOCATION ENDED   VMSEXT allocation complete.

ENTER FORMAT OR ALLOCATE:                  No reply necessary to this response.

FORMAT                                     Begin formatting VMPK01.

FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
125
ENTER DEVICE TYPE:
3330-11
ENTER START CYLINDER (XXX) OR "LABEL":
000
ENTER END CYLINDER (XXX):
807
ENTER DEVICE LABEL:
VMPK01
FORMAT STARTED

FORMAT DONE                                VMPK01 formatting complete.
```

```
000 NO. PAGE RECORDS WITH READ-CHECK ERRORS
ENTER FORMAT OR ALLOCATE:
```

No reply necessary to this response.

```
ALLOCATE
```

Begin allocating VMPK01.

```
ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
125
ENTER DEVICE TYPE:
3330-11
ENTER DEVICE LABEL:
VMPK01
ENTER ALLOCATION DATA FOR VOLUME VMPK01
TYPE CYL CYL
.... ... ...
PERM 000 119
TDSK 120 214
TEMP 215 334
PERM 335 807
END
ALLOCATION RESULTS
PERM 000 119
TDSK 120 214
TEMP 215 334
PERM 335 807

DEVICE 125  VOLUME VMPK01 ALLOCATION ENDED
```

VMPK01 allocation complete.

```
ENTER FORMAT OR ALLOCATE:
```

No reply necessary to this response.

Formatting and allocating the work disks are now complete. To clear old data and make the results of PREP CPFMT allocation known to CP, issue:

```
ATTN
```

On a 3278 or 3279 terminal device, press PA1.

Warning: When running a *second level* machine, the *first level* linend character # has to be changed to some other character. Enter TERM LINEND % if you did not do so previously.

Then, enter:

- At *first level*, **#cp ipl 190**
- At *second level*, **#cp ipl 190 parm autocr**

```
VM/SP n STARTER
```

ENTER

This entry only necessary at *first level.*

```
R;
```

Issue **detach 124** if VMSEXT was just formatted in this step. (124 is a virtual address.) Only needed for 3310, 3330, or 3340 devices.

detach 124

```
DASD ' CUU1 ' DETACHED MAINT 124
R;
```

Issue **detach 125** if VMPK01 was just formatted in this step. (125 is a virtual address.)

detach 125

```
DASD ' CUU1 ' DETACHED MAINT 125
R;
```

attach cuu1 system vmpk01

cuu1 is the real device address of your extra work volume, VMPK01.

```
DASD ' CUU1 ' ATTACH TO SYSTEM VMPK01
R;
```

attach cuu2 system vmsext

cuu2 is the real device address of your work volume, VMSEXT. Only needed for 3310, 3330, or 3340 devices.

```
DASD ' CUU2 ' ATTACH TO SYSTEM VMSEXT
R;
```

B. PREP CMSFMT - Formats Minidisks for Receiving CP/CMS Files

You will now use the CMSFMT option of PREP EXEC to load sample files and allocate minidisk space for CP/CMS files. The PREP CMSFMT command automatically invokes the PREP LOAD option. It loads all files from the Product Tape (unless the ONLY option is specified).

acc 191 c

This EXEC runs from the C minidisk.

```
' 191 A ' RELEASED
R;
```

prep cmsfmt

If you do not access the 191 minidisk as shown above, ignore any related disk accessing messages.

```
' CMSFMT ' PROCESS STARTING ...

LOAD THE SYSTEM DEFINITION SAMPLES.
DEFAULT FILES LOADED ON MAINT'S 191 DISK

FORMATTING MAINT 194 MINIDISK  (CP  TEXT DECK DISK)
FORMATTING MAINT 201 MINIDISK  (EREP LIBRARY)
FORMATTING MAINT 19D MINIDISK  (CMS HELP DISK)
FORMATTING MAINT 293 MINIDISK  (CMS SERVICE DISK)
FORMATTING MAINT 294 MINIDISK  (CP  SERVICE DISK)

' CMSFMT ' EXECUTION COMPLETE.
```

Note: If you get failures while formatting the minidisks, you can restart with Part B of this step.

This step continues on the next page without user intervention.

C. PREP LOAD - Loads Product Tape to Disk

For systems operating at *second level*, this portion of Step 6 may require considerable time to process if the system load is high.

Note: This procedure automatically generates the system ASSEMBLER command.

```
' LOAD ' PROCESS STARTING . . .

LOADING   MAINT 194 MINIDISK (CP TEXT DECK DISK)
MAINT 194  LOADED

LOADING   MAINT 19D MINIDISK (HELP FILE DISK)
MAINT 19D  LOADED

LOADING MAINT 190 MINIDISK (CMS FILE DISK)
190 ALSO = S-DISK
MAINT 190 LOADED

THE PRODUCT TAPE HAS BEEN LOADED TO DISK. IF SOURCE IS
NOT TO BE LOADED TO DISK, YOU MAY WISH TO REPLACE
THE PRODUCT TAPE WITH A SCRATCH TAPE AT THIS POINT.

' LOAD ' EXECUTION COMPLETE.
GENERATING THE SYSTEM ASSEMBLER COMMAND.
NO RESPONSE IS REQUIRED FOR THE ASSEMBLE PROMPTS.

ASSEMBLE XF GEND PROC

ENTER TARGET DISK MODE FOR ASSEMBLE MODULES
DEFAULTS TO S-DISK IF NONE ENTERED.
ASSEMBLE XF GEND COMPLETE

**** IPL 190 PARM AUTOCR TAKING PLACE ****
VM/SP n STARTER
R;
```

You have installed the Product Tape at this point. It can now be removed. If you wish to execute optional PREP EXEC tasks or install service updates, proceed to Phase 4 on the next page. Otherwise, you may proceed to Phase 5 on page 53.

Phase 4. Executing Optional Steps

Phase Overview

In Phase 4 you will be doing:

Step 7. Execute optional PREP EXEC tasks.

Step 8. Apply service updates, if any.

| The instructions for completing these steps are on the pages that follow.

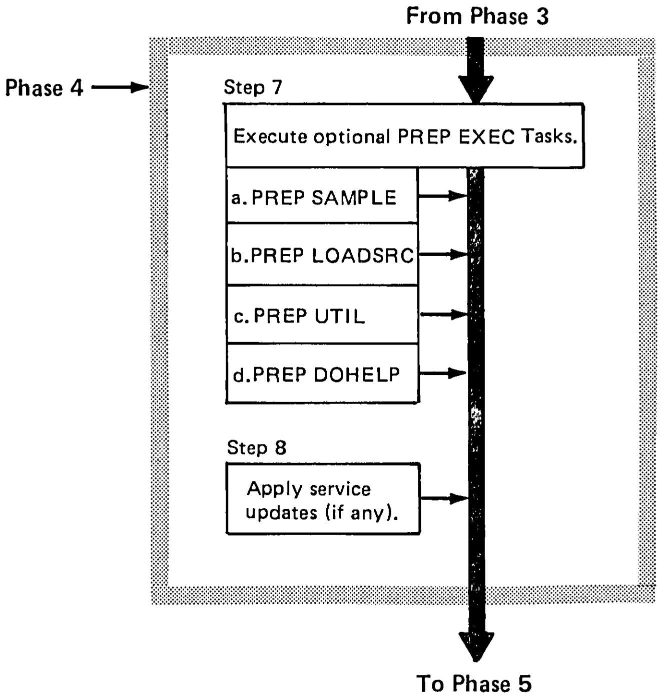


Figure 9. Optional Tasks (Starter System)

Step 7. Perform Optional PREP EXEC Tasks

During the installation process you might need to tailor some sample files shipped with the system. You may also want to load system source code or load utility programs. There are PREP EXEC options to assist you. They are:

A. PREP SAMPLE - Print out copies of sample files.
B. PREP LOADSRC - Load CP and/or CMS source code.
C. PREP UTIL - Create stand-alone utility program tape.
D. PREP DOHELP - Erase HELP files or convert them to uppercase.

A. PREP SAMPLE: Print Hard Copy of Sample Files

To spool to the system printer copies of

DMKSYS ASSEMBLE
DMKSNT ASSEMBLE
DMKRIO ASSEMBLE
VMUSERS DIRECT
MDISKMAP

enter:

```
  prep sample

WOULD YOU LIKE THE ' SAMPLE ' FILE PRINTED? ( YES | EXIT )

  ENTER                                         The null entry defaults to yes.

THE FOLLOWING FILES HAVE BEEN SPOOLED TO THE SYSTEM PRINTER
DMKSYS ASSEMBLE
DMKSNT ASSEMBLE
DMKRIO ASSEMBLE
VMUSERS DIRECT
MDISKMAP 'dasdtype'

' SAMPLE ' EXECUTION COMPLETE.
R;
```

B. PREP LOADSRC: Load CP and/or CMS Source Code

This part of Step 7 assumes that formatting for volumes.

- VMSTGE (CP and CMS source codes except for 3310 and 3340 devices)
- VMSTG2[9] (CMS source code for 3310 and 3340 devices)

has/have been completed. Also, your selection is labeled and online in preparation for loading. If not, skip to the next page. To initiate loading of source code issue:

prep loadsrc

```
' LOADSRC ' PROCESS STARTING ...
INDICATE SOURCE TO BE LOADED: ( BOTH | CP | CMS | EXIT )
DEFAULT IS "BOTH" (NULL RESPONSE)
```

both

Loads both CP and CMS source.

```
'B (393) ' DEVICE ERROR
IN ORDER TO LOAD THE SOURCE CODE,
DISK 393 MUST BE FORMATTED.
MINIDISK 393 IS NOT FORMATTED.
DO YOU WISH TO FORMAT IT ? ( YES | EXIT )
```

Ignore this message.

yes

```
FORMATTING MINIDISK 393 ...
```

```
'B (394) ' DEVICE ERROR
IN ORDER TO LOAD THE SOURCE CODE,
DISK 394 MUST BE FORMATTED.
MINIDISK 394 IS NOT FORMATTED.
DO YOU WISH TO FORMAT IT ? ( YES | EXIT )
```

Ignore this message.

yes

```
FORMATTING MINIDISK 394 ...

IF THE PRODUCT TAPE (1) YOU RECEIVED IS 6250 BPI, THE SOURCE CODE
IS LOCATED ON FILES 6 AND 7 OF THIS TAPE.

IF THE PRODUCT TAPES (3) YOU RECEIVED ARE 1600 BPI, THE SOURCE CODE
IS ON TWO OF THESE TAPES.  THE APPROPRIATE SOURCE CODE
TAPE MUST BE MOUNTED AND ATTACHED TO YOUR ID (EG. MAINT) AS VIRTUAL
181.

IS THE SOURCE CODE ON A 1600 OR 6250 BPI TAPE ?
( 1600 | 6250 | EXIT )
```

1600

```
LOADING CP SOURCE FILES ONTO 394 MINIDISK
CP SOURCE FILES LOADED

REMOVE THE CP SOURCE TAPE AND MOUNT THE CMS SOURCE TAPE
ON THE SAME DRIVE AND PRESS "ENTER" WHEN READY.

LOADING  CMS SOURCE FILES ONTO 393 MINIDISK
CMS SOURCE FILES LOADED

' LOADSRC ' COMPLETE.
R;
```

[9] VMSTG2 must be formatted using the FORMAT/ALLOCATE program. This can not be done while operating under a starter system. Do this step after the new CP has been loaded.

C. PREP UTIL: Create Utility Routine Tape

In this step you create a tape containing one or more of the following stand-alone service routines:

Device Support Facilities Program
Directory Program
VM/SP FORMAT/ALLOCATE Program
DASD Dump Restore (DDR) Program

To begin, issue:

prep util

```
A SCRATCH TAPE WITH A RING SHOULD
BE MOUNTED AS VIRTUAL TAPE 181

' UTIL ' PROCESS STARTING ...
PRESSING THE "ENTER" KEY FOR THE FOLLOWING QUESTIONS
IS THE DEFAULT FOR A "NO" RESPONSE.
```

The sample sequence that follows shows how you can select the service routines to be placed on tape. The first two service programs, Device Support Facilities and Directory, are not placed on tape. However, the last two, VM/SP FORMAT/ALLOCATE and DDR, are placed on the utility tape.

A null response defaults to **no** for each of the four prompts issued here.

```
DO YOU WISH THE DEVICE
SUPPORT FACILITIES PROGRAM
TO BE PLACED ON TAPE ?
( NO | YES | EXIT )
```

ENTER

```
DO YOU WISH THE DIRECTORY PROGRAM TO BE PLACED ON TAPE ?
( NO | YES | EXIT )
```

ENTER

```
DO YOU WISH THE FORMAT/ALLOCATE PROGRAM TO BE PLACED ON TAPE ?
( NO | YES | EXIT )
```

yes

```
MOVING ' IPL FMT ' TO TAPE . . .
' IPL FMT ' IS NOW ON TAPE AS FILE NUMBER 1

DO YOU WISH THE DDR PROGRAM TO BE PLACED ON TAPE ?
( NO | YES | EXIT )
```

yes

```
MOVING ' IPL DDR ' TO TAPE . . .
' IPL DDR ' IS NOW ON TAPE AS FILE NUMBER 2

THE PROGRAMS ARE NOW LOADED ON TAPE.

WHEN THE TIME COMES TO USE THE TAPE, AN IPL OF THE TAPE IS
REQUIRED TO REACH EACH FILE OF THE TAPE.
ERROR MESSAGES ENCOUNTERED BETWEEN CERTAIN IPL'S ARE DUE TO
PROGRAM TERMINATION DIFFERENCES AND MAY REQUIRE AN ADDITIONAL IPL.

' UTIL ' IS COMPLETE.
R;
```

D. PREP DOHELP: Help file management

The PREP DOHELP option allows you to erase HELP files or translate them to uppercase. The HELP files on the CMS system disk extension are in mixed lowercase and uppercase character representation. In some installations, lowercase characters are reserved for display of special alphabets. In such installations, HELP files should be displayed in uppercase representation only.

To initiate this option, enter:

prep dohelp

```
' DOHELP ' PROCESS STARTING ...
ENTER "ERASE" TO ERASE ALL HELP FILES ON THE SPECIFIED DISK.
ENTER "UP"    TO TRANSLATE ALL HELP FILES TO UPPERCASE REPRESENTATION.
ENTER "EXIT"  TO EXIT.
```

up

```
THE HELP FILES TO BE TRANSLATED TO UPPER CASE ARE
ASSUMED TO RESIDE ON DISK 19D.  IF THEY RESIDE ELSEWHERE,
ENTER DISK (E.G. 191, 192, 19C. ETC.).
PRESS ENTER FOR DEFAULT. TYPE EXIT TO EXIT.
```

ENTER

```
'19D' REPLACES ' B (190) '
R;
```

If you have moved the CMS system disk to a disk other than the one designated by the Starter System, you can translate the HELP files to uppercase character representation on your new minidisk and keep the Starter System as it is.

Proceed to Step 8 on the next page.

Step 8. Apply Service Updates

Note: Any PUT tape *Memo to Users* information replaces or supplements information in this step. Be sure to review it before applying service.

A VM/SP service tape might be supplied with VM/SP. This tape contains cumulative service for VM/SP. You will now use the two service minidisks, 293 and 294, that you formatted in Step 6(B) using PREP CMSFMT.

Mount the service tape (with no ring) as virtual drive 181 and issue the following commands:

```
attach cuu * 181
```

cuu is the real address of the service tape device.

```
access 191 c

vmfplc2 load * * c

vmserv
```

The VMSERV EXEC maps the service tape and allows you to print the PUT document and *Memo to Users.*

```
DO YOU WANT TO PRINT THE MEMO TO USERS?

  yes
```

VMSERV then prints the PUT document and *Memo to Users*, and reminds you to read them prior to installing service. After reviewing the documentation and contacting IBM concerning the latest service activity, you can install service as described in the PUT document and *Memo to Users.*

Do *not* build a new CP or CMS nucleus as described in the user memo, but continue here after applying service from the PUT. After applying service, recreate the ASSEMBLE module as follows:

```
access 190 c

asmgend

ipl 190 parm autocr
```

Proceed to Phase 5 on the next page.

Phase 5. Build the CP and CMS Nuclei

Note: From this point on, you are customizing the system to your installation's requirements. Therefore, the instruction that you are given contain general information rather than detailed instructions.

Phase Overview

In Phase 5 you will be doing:

Step 9. Modify System Installation-Dependent Files

Step 10. Generate the CP Nucleus

Step 11. Generate the CMS Nucleus

The instructions for completing these steps are on the pages that follow.

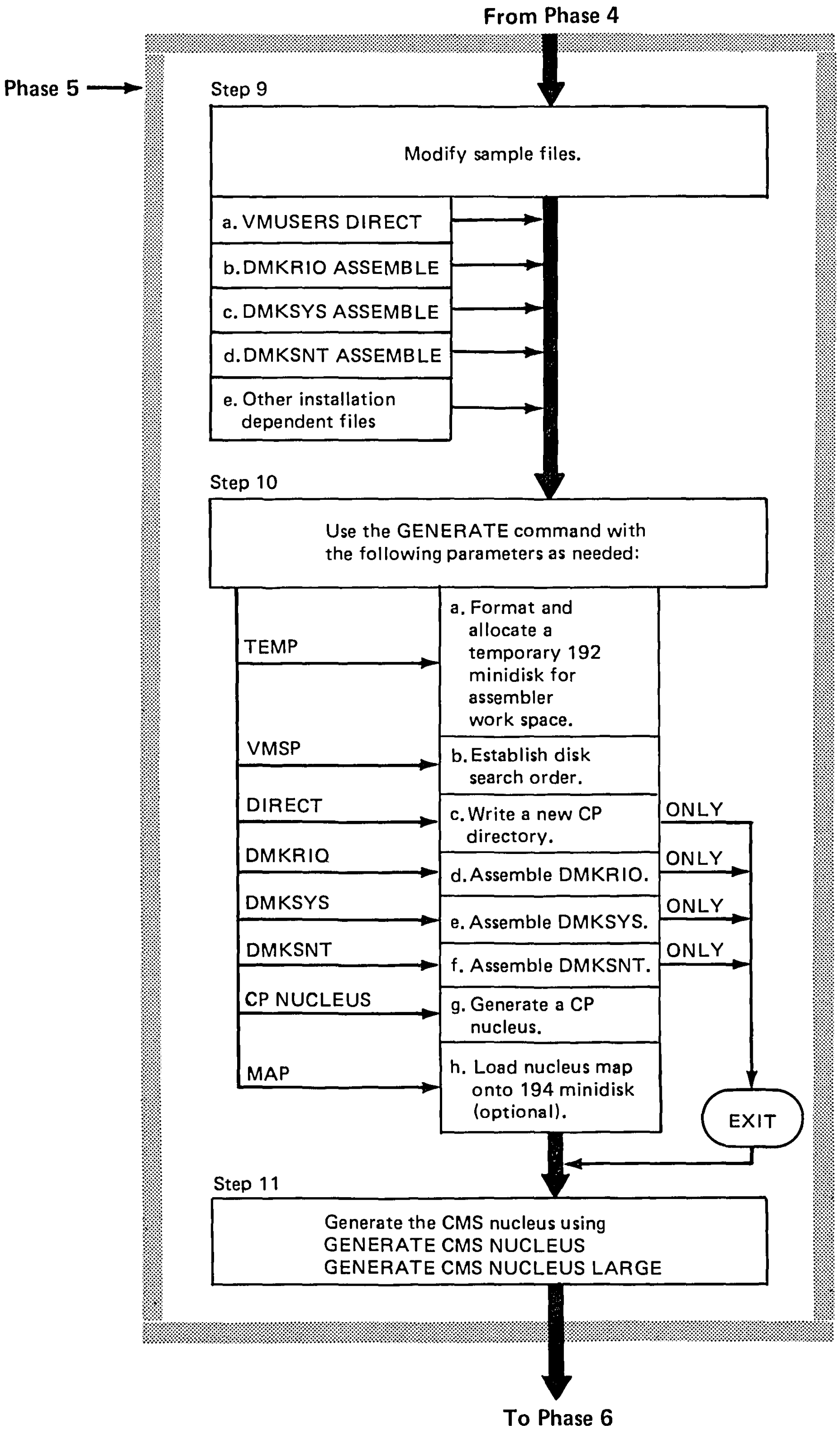

From Phase 4
Phase 5
Step 9
Modify sample files.
a. VMUSERS DIRECT
b. DMKRIO ASSEMBLE
c. DMKSYS ASSEMBLE
d. DMKSNT ASSEMBLE
e. Other installation dependent files
Step 10
Use the GENERATE command with the following parameters as needed:
TEMP
a. Format and allocate a temporary 192 minidisk for assembler work space.
VMSP
b. Establish disk search order.
DIRECT
c. Write a new CP directory.
ONLY
DMKRIO
d. Assemble DMKRIO.
ONLY
DMKSYS
e. Assemble DMKSYS.
ONLY
DMKSNT
f. Assemble DMKSNT.
ONLY
CP NUCLEUS
g. Generate a CP nucleus.
MAP
h. Load nucleus map onto 194 minidisk (optional).
EXIT
Step 11
Generate the CMS nucleus using
GENERATE CMS NUCLEUS
GENERATE CMS NUCLEUS LARGE
To Phase 6

Figure 10. System Definition and Generation

Warning: The GENERATE EXEC changes your reader, printer and punch to Class I. In some instances the class might not be reset. If later work requires a class other than Class I, reset the class after executing PREP. For example, issue CHANGE PRINTER CLASS c1 where c1 is the selected spool class. Similar commands for your reader and punch also should be issued.

Step 9. Modify System Installation-Dependent Files

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied with VM/SP serve as a base for building files unique to your installation. You may need to modify one or more of these files now.

Examine the listings of the Sample Directory and DMK*xxx* files which are reproduced in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

Use the System Product Editor to modify your sample files if desired. For more information, see "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference.*

Directory Considerations

The VM/SP SAMPLE directory is the VMUSERS DIRECT file loaded in Step 6.

If you are installing optional feature program products, you may need to modify the directory (VMUSERS DIRECT); for example, because of space requirements. Directory related information for each of the optional feature program products is shown in Appendix I, "Optional Feature Program Products" on page 287. You can update this file later to include additional virtual machine definitions and userids. See "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference* for a general discussion. For more detailed information, see "A Virtual Machine for Updating VM/SP" on page 142.

If you made changes to the directory, you may want to use the DISKMAP EXEC to check for erroneous overlaps.

DMKRIO Considerations

DMKRIO defines the real I/O configurations. If the default DMKRIO file does not match your system installation, you need to alter it now, before you build your new CP nucleus. A copy of the DMKRIO file can be found in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

For more information about modifying your own DMKRIO, see "Chapter 19. Preparing the Real I/O Configuration File (DMKRIO)" in the *VM/SP Planning Guide and Reference.*

DMKSNT Considerations

If you plan to add products, other than optional feature program products that require a saved segment, the SNT entries should be added now. This prevents later reassembly of DMKSNT and CP nucleus regeneration. Check user-selected product installation documentation for SNT related information. A description of CMS saved segment relocation can be found in "Planning for CMS" in Part 1 of the *VM/SP Planning Guide and Reference.*

For the procedure regarding alternate CMS nucleus placement, see Appendix E, "Example of Alternate CMS Nucleus Placement" on page 275. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for an example to modify the CMS nucleus to include file status tables for the S- and Y-minidisks.

DMKSYS Considerations

No space dedicated to paging has been provided in the system samples. An additional volume is required for paging space. The new volume id should be added to the SYSOWN macro definition to contain the paging space. Refer to "Direct Access Storage Requirements for CP" in the *VM/SP Planning Guide and Reference* for more information.

During execution of the GENERATE EXEC process, you will be given the opportunity to write the CP nucleus to tape. Be aware that the SYSRES=123 definition in the sample DMKSYS file requires a real address of 123, if this was to be used as a standalone backup tape. If you wish to change this to a real address, you must also change the corresponding DIRECT MDISK 123 statement for the MAINT userid and the DIRECTORY 123 statement in the sample directory.

Other Considerations

The following files can be altered as required to meet your installation's needs.

- Printer Universal Character Set[10]

  DMKUCS (1403)
  DMKUCB (3211)
  DMKUCC (3203)
  DMKPIB (3262)

- Font Offset Buffer[10]

  DMKPIA (3289)

- Forms Control Macros[10]

  DMKFCB (forms control)

- LOGO

  DMKBOX

Proceed to Step 10.

[10] See "Updating Printer Modules" on page 192 for further information.

Step 10. Generating a CP Nucleus

The procedure for generating a CP nucleus is managed by GENERATE EXEC. GENERATE EXEC provides entry points for generating selected portions of the CP nucleus. Figure 10 on page 54 shows the entry points and exits of the sequence.

An entry point is selected by specifying the appropriate operand following the GENERATE command. For instance, placing the TEMP operand after GENERATE causes a complete CP nucleus to be created.

Note: If you plan to build the CP nucleus using GENERATE CP NUCLEUS instead of GENERATE TEMP, you must assemble all of the DMKRIO, DMKSYS, and DMKSNT files prior to entering the command.

Four of the operands allow an ONLY option. If "ONLY" is specified after the DIRECT, DMKRIO, DMKSYS, or DMKSNT operand, *only* the GENERATE code associated with the specific operand is executed.

If you would like to display HELP information that describes any GENERATE entry point, enter:

generate help For more detailed information on the GENERATE EXEC, see "GENERATE" on page 197.

If you are using one tape drive only, issue:

def 181 182 Allows you the option to save the CP nucleus on tape later in this step.

Replace the tape currently mounted with a scratch tape (with ring) to receive the nucleus code.

If you plan to include the Small CP or Virtual=Real option, read Appendix G, "Special Options for CP" on page 283 before proceeding.

Step 10

This procedure assumes that you enter GENERATE EXEC from the top. To start, get temporary disk space by entering:

generate temp

The TEMP option causes execution of the entire GENERATE EXEC. The GENERATE TEMP option formats and allocates temporary minidisk 192 (B-disk) for assembler workspace. If an error occurs, the routine exits at the step being processed.

```
GET TEMP SPACE ON DASD TYPE = 3330
DISK 'B' NOT ACCESSED
DEV 192 DOES NOT EXIST
DASD 192 DEFINED
FORMAT - TEMP 30 CYLINDER MDISK.
TEMP MDISK 192 ACCESSED AS FILE MODE B.
```

Executing GENERATE TEMP code.

```
DO YOU WISH TO UPDATE THE SYSTEM
DIRECTORY AND ASSEMBLE THE SYSTEM
DEFINITION FILES NOW?
RESPOND ( YES | NO )
```

yes

yes means that RIO, SNT, and SYS will be assembled and the directory file will be placed online.
no results in only a temporary minidisk being allocated and an exit from GENERATE EXEC.

```
'194' REPLACES ' A (191) '
C (191) R/O
ASSEMBLE FILE SEARCH ORDER IS 194 THEN 191.
```

Executing GENERATE VMSP code.

```
ENTER DIRECTORY FILENAME:
THE NULL DEFAULT VALUE IS 'VMUSERS'.
TO BYPASS STEP ENTER 'SKIP' -
(GO TO DMKRIO ASSEMBLE STEP)
```

Executing GENERATE DIRECT code.

ENTER

VMUSERS is now your directory filename.

```
EOJ DIRECTORY UPDATED

WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:    ( NO | YES )
```

no

This sample procedure assumes a uniprocessor system.

```
VMFASM BEING EXECUTED WITH
FN= DMKRIO CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKRIO
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKRIO TEXT CREATED
```

Executing GENERATE DMKRIO code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSYS CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSYS
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSYS TEXT CREATED
```

Executing GENERATE DMKSYS code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSNT CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSNT
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSNT TEXT A1 CREATED
'194' REPLACES ' A (194) '
```

Executing GENERATE DMKSNT code.

Before you go on, the GENERATE TEMP command and the functions it performs must complete successfully. If any errors occurred while the directory was being built or while the DMKxxx files were assembling, the directory program and the GENERATE EXEC procedure issued the message:

```
EDIT AND CORRECT THE DIRECTORY FILE IN ERROR.

                    - or -

EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR.
```

If you received one of these messages, read the next page for the procedure to correct the error condition. Otherwise, continue on page 61.

Read this page if you received one of the error messages on the previous page.

ERROR CONDITIONS

A. "EDIT AND CORRECT THE DIRECTORY FILE IN ERROR" Message

Errors were detected while the Directory was being loaded.

The DIRECT Option reads the directory file from your minidisk and writes it to the system residence volume. Specifying GENERATE DIRECT ONLY causes an exit from the GENERATE routine after the new directory is written. If "ONLY" is not specified, the routine continues with assembly of DMKRIO, DMKSYS, and DMKSNT. You need to edit and correct the Directory file using the System Product Editor. Then, respond with:

generate direct

B. "EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR" Message

Errors were detected during assembly of one or more of the DMKRIO, DMKSYS, or DMKSNT files.

Use the System Product Editor to correct the indicated "filename" (DMKRIO, DMKSYS, or DMKSNT). Then, enter the GENERATE command with the appropriate option. See the following example:

generate dmkxxx

dmkxxx represents **dmkrio, dmksys, or dmksnt.**

```
WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:    ( NO | YES )
```

This message appears only when generating DMKRIO.

no

This sample procedure assumes a uniprocessor system.

```
VMFASM BEING EXECUTED WITH
FN= DMKxxx CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKxxx
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKxxx TEXT CREATED
```

- If **dmkxxx** = **dmkrio**, then GENERATE DMKRIO is executed. The DMKRIO Operand assembles DMKRIO. If the ONLY option is not specified, it also:

  1. Assembles DMKSYS and DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksys**, then GENERATE DMKSYS is executed. The DMKSYS Operand assembles DMKSYS. If the ONLY option is not specified, it also:

  1. Assembles DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksnt**, then GENERATE DMKSNT is executed. The DMKSNT Operand assembles DMKSNT. If the ONLY option is not specified, it also generates a CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete.

```
DO YOU WANT THE SMALL CP OPTION? --
ENTER:  ( NO | YES )
```

See Appendix G, "Special Options for CP" on page 283 if you want the Small CP option.

no

Executing GENERATE CP NUCLEUS. The CP NUCLEUS operand causes the CP nucleus to be generated. A CP nucleus can be written to tape or to the target disk. If the NOLOAD option is specified, the resulting nucleus will not be IPLed automatically.

```
WHICH ARE YOU GENERATING: A
'UNI PROCESSOR' (UP),
AN 'ATTACHED PROCESSOR' (AP),
OR A 'MULTI PROCESSOR' (MP) SYSTEM ?

ENTER: ( UP | AP | MP )
```

Your responses to these questions determines the CNTRL file and LOADLIST EXEC used to correctly build your system.

up

```
VIRTUAL=REAL OPTION REQUIRED (YES|NO):
```

See Appendix G, "Special Options for CP" on page 283 if you want the V=R option.

no

```
NO FILES PURGED
VMFLOAD BEING EXECUTED
LOADLIST= CPLOAD AND CNTRL=DMKSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT  COPY 001  NOHOLD

AN IPL'ABLE CP NUCLEUS EXISTS IN YOUR VIRTUAL CARD READER.
DO YOU WISH TO CREATE AN IPL'ABLE TAPE
OR, LOAD THE NUCLEUS IN YOUR READER DIRECTLY
TO YOUR SYSTEM RESIDENCE DISK VOLUME?
ENTER: ( DISK | TAPE | EXIT ).
```

tape

You are saving the new CP nucleus on tape as well as on DASD. If any errors are detected while the tape is being written, you must recreate the CP nucleus. To do this, enter GENERATE CP NUCLEUS. The procedure then restarts at the point above where you are asked: `DO YOU WANT THE SMALL CP OPTION? --`

```
IPLABLE NUCLEUS NOW ON TAPE ****

WHEN YOU RECEIVE THE MESSAGE: NUCLEUS LOADED ON 'volid'
-ENTER: 'IPL 190 PARM AUTOCR'
TO PUT THE CP LOAD MAP ONTO YOUR 194 MINIDISK, AFTER THE IPL
-ENTER: GENERATE MAP

NUCLEUS LOADED ON VMSRES - STARTING CYL/BLK=399,
LAST CYL/BLK USED=402
```

The message below does not automatically display on many processors. Check your processor reference manual for information about displaying and interpreting the PSW. If the PSW message is not displayed, press PA1 to get a CP READ state, then proceed.

```
CP ENTERED; DISABLED WAIT
PSW '0002 0000 0000 0012'
```

This is the normal return if the nucleus loaded correctly.

If you do not receive the **NUCLEUS LOADED ON 'volid'** message, read the following procedure to correct the error condition. Otherwise, continue on page 63.

ERROR CONDITION

No "NUCLEUS LOADED ON 'volid' " Message

- Inspect the error load map in the virtual reader and the virtual PSW. A loader error may be indicated on the listing or PSW. See the *VM/SP System Messages and Codes* for a list of the loader wait state codes.

  After correcting the error, issue GENERATE CP NUCLEUS.

- If you shut down following a nucleus loading error, you may not be able to reIPL the system. If so:

  1. Mount the Starter System Tape if it is not already attached.
  2. IPL the Starter System Tape *twice* to load the DDR program.
  3. Follow Step 2 of this procedure, starting on page 21, through the first three commands that define input and output devices. Then, enter RESTORE commands, with parameters appropriate for your DASD type, as follows:

     For FB-512 (3310/3370) devices:

restore 00000 00015	Restore Starter System allocation.
restore 92480 94303	Restore Starter System directory.
restore 123736 126015	Restore Starter System nucleus.

For CKD devices:

restore 000 000	Restore Starter System allocation.
restore aaa bbb	Restore Starter System directory.
restore xxx yyy	Restore Starter System nucleus.

Where aaa bbb and xxx yyy are:

Device Type	**aaa**	**bbb**	**xxx**	**yyy**
3330	275	278	397	403
3340	070	079	032	043
3350	101	102	549	554
3375	144	147	953	958
3380	097	098	881	884

You can now reIPL, recreate your directory using GENERATE DIRECT ONLY, and generate a new CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete. To reload CMS, enter:

ipl 190 parm autocr

For *second level* operation, **ENTER** after IPLing.

```
VM/SP n STARTER
R;
```

generate map

Executing GENERATE MAP to load the CP nucleus map onto the 194 minidisk.

```
LOAD MAP HAS BEEN PLACED IN READER


SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cp

```
'194' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CPNUC MAP" ON MDISK "194"

R;
```

The contents of the load map are described in "Chapter 9. Updating An Installed VM/SP System" on page 165. Two external names may be listed as undefined on the load map. The external name DMKSLC is undefined if the Virtual=Real option is not selected. The external name DMKRNTBL is undefined if there is no entry in the system name table for a 3704/3705 control program. (Provided you did not code a NAMENCP macro in the DMKSNT file.) Also, other names might be listed as undefined if other modules were deleted as described in "Reducing the CP Nucleus Size" in Part 1 of the *VM/SP Planning Guide and Reference.*

You have now generated the CP nucleus having had the chance to select Small CP or V=R options. At this point continue with Step 11.

Step 11. Generating a CMS Nucleus

To initiate this step, enter:

generate cms nucleus

The exact text of the first message issued by GENERATE CMS NUCLEUS depends on the device type being used for the CMS system disk (S-disk).

```
' 190 ' REPLACES ' A(194) '
190 ALSO S-DISK

3330 IS THE "S DISK" DASD TYPE.
IF SAMPLE SIZES ARE USED, REPLY TO THE PROMPT "CYLBLK NUMBER":
WITH ' 94 ' IF THE CMS NUCLEUS IS TO BE GENERATED OR
WITH ' 97 ' IF THE CMSL NUCLEUS IS TO BE GENERATED
NO FILES PURGED


VMFLOAD BEING EXECUTED
LOADLIST= CMSLOAD AND CNTRL= DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn  TO  MAINT COPY 001   NOHOLD

SAVE A COPY OF THE CMS NUCLEUS AS A DISK FILE?
ENTER: ( NO | YES )
```

yes

```
THE CMS NUCLEUS NOW EXISTS ON DISK AS FILE ' CMSNUC NUCLEUS A1 '.

THE LOADMAP FOR THE CMS NUCLEUS SHOULD BE SAVED ON MDISK 191.
WHEN THIS STEP COMPLETES, THE CMS SYSTEM IS RE-IPLED AND GENERATED.

CMS LOAD MESSAGES APPEAR NEXT.
RESPOND AS INDICATED.
```

See “Chapter 9. Updating An Installed VM/SP System” on page 165for an explanation of the prompts and responses.

```
DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

190

This device address is the virtual address where the CMS nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =

    94
```

The nucleus will reside on the last cylinder(s) or block(s) of the 190 minidisk. Your S-disk DASD type is displayed in the first message in Step 11. (3330 in this sample.) Respond with the cylinder/block number also displayed in that message. (**94** in this sample.) Use the accompanying chart for the values you need for your device type.

Device	CMS	CMSL
FB-512	43520 blk	44544 blk
3330	094 cyl	097 cyl
3340	222 cyl	229 cyl
3350	045 cyl	047 cyl
3375	070 cyl	072 cyl
3380	043 cyl	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?

    yes

DMSINI611R VERSION IDENTIFICATION =

    ENTER
```

You are permitted to enter up to 32 descriptive characters.

```
DMSINI612R INSTALLATION HEADING =

    ENTER
```

You are permitted to enter up to 64 descriptive characters.

```
VM/SP REL n  mm/dd/yy hh:mm

    ENTER
```

The new CMS nucleus is automatically reIPLed by ENTER.

```
R;

    generate map

LOAD MAP HAS BEEN PLACED IN READER


SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END

    cms

'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSNUC MAP" ON MDISK "191"
R;
```

Proceed to Phase 6 on the next page.

Phase 6. Conclude the Sample System Installation

Phase Overview

In Phase 6 you will be doing:

Step 12. IPL the newly generated system.

Step 13. Format the operator's 191 minidisk.

Step 14. Updating the SYSTEM NETID.

Step 15. Save CMS.

Step 16. Create a backup system on tape (optional).

| The instructions for completing these steps are on the pages that follow.

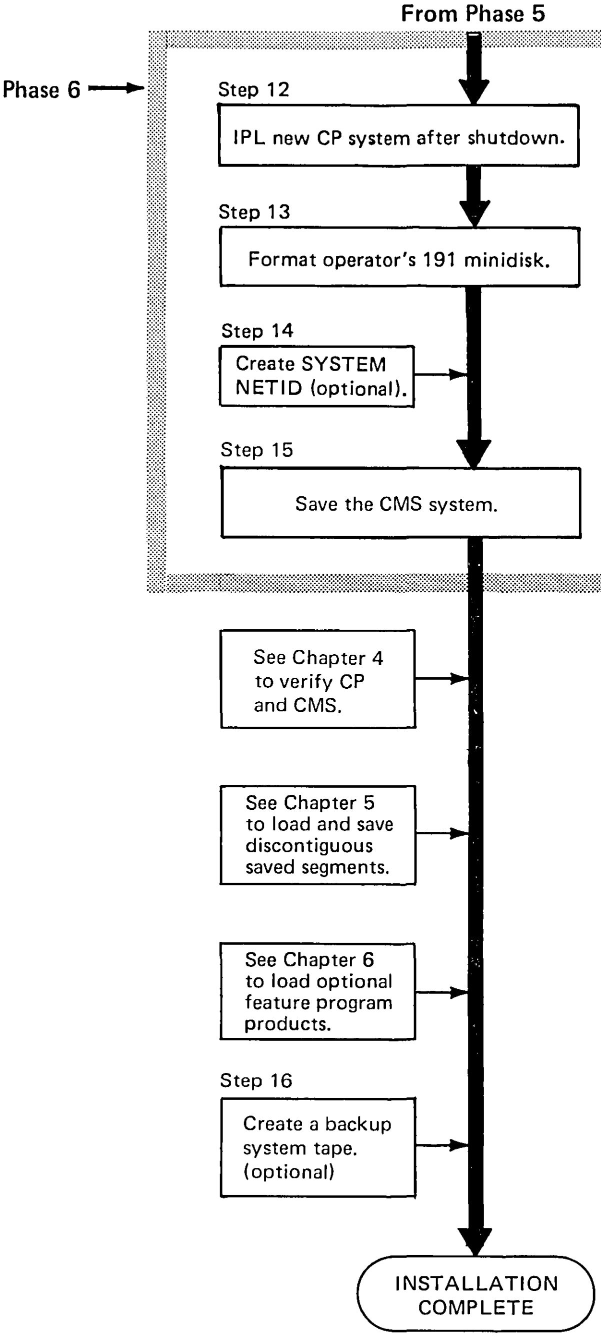

From Phase 5
Phase 6
Step 12
IPL new CP system after shutdown.
Step 13
Format operator's 191 minidisk.
Step 14
Create SYSTEM NETID (optional).
Step 15
Save the CMS system.
See Chapter 4 to verify CP and CMS.
See Chapter 5 to load and save discontiguous saved segments.
See Chapter 6 to load optional feature program products.
Step 16
Create a backup system tape. (optional)
INSTALLATION COMPLETE

Figure 11. Conclude Sample System Installation

Step 12. IPL the Newly Generated System

In this step, you IPL your generated VM/SP system rather than the starter system.

Be aware that the output is spooled class T. If you want the output to be printed, you must alter the class of the output or the printer.

First, spool your console closed by issuing:

```
   sp con close
R;
```

Second, drain all spooling devices by entering:

```
   drain all
R;
```

Be sure to wait until all attached I/O devices are drained as indicated by the **R;** ready message.

Then, shutdown the Starter System by entering:

```
   shutdown
```

The system operator userid, assigned in DMKSYS (SYSOPR macro), is logged on when you IPL the newly created system residence volume.

- At *first level*, IPL (Initial Program Load) the new system using the real address of VMSRES. Follow the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)
- At *second level*, define VMSRES at virtual address 123 if not previously done.

  def cuu 123 — **cuu** is the present virtual address of your VMSRES volumes.

  Then, IPL the virtual address of VMSRES.

  ipl 123 — **123** for this sample.

```
VM/SP REL n, SERVICE LEVEL 0000

NOW hh:mm:ss EDT day mm/dd/yy
CHANGE TOD CLOCK (YES|NO):
```

At *first level*, refer to your hardware documentation. At *second level*, always reply **no**.

```
   no

START ((COLD|WARM|CKPT|FORCE)
(DRAIN))|(SHUTDOWN):
```

Because there is no data or accounting information to be recovered, you can request a cold start.

```
   cold
```

```
AUTO LOGON *** OPERATOR USERS=001 BY SYSTEM

DMKCPI952I nnnnK SYSTEM STORAGE
DMKCPI957I STOR sssssK, NUC nnnK, DYN dddddK,
TRA tttK, FREE ffffK, V=R vvvvvK

FILES:  NO RDR,  NO PRT,  NO PUN
FORMATTING ERROR RECORDING AREA

INITIALIZATION COMPLETE
```

When you IPL the newly generated system, your system residence volume is defined by specifications of the SYSRES macro. If you used the directory supplied with the Starter System, the Starter System volume (VMSRES) is set up as entered in Step 3.

Proceed to Step 13.

Step 13. Format the Operator's 191 Minidisk

Before any new minidisk area can be used for CMS files, it must be initialized. Use the CMS FORMAT command to format the area into fixed-sized blocks. The CMS FORMAT command is described in *VM/SP CMS Command and Macro Reference.*

At this time you are logged on as the operator. If you have not already loaded CMS, enter:

```
   ipl 190
```

When you give the IPL 190 command, you may receive a device error message. You can ignore this message if it is received.

```
VM/SP REL n MM/DD/YY HH:MM

   access (nodisk
```

The NODISK option prevents CMS from automatically accessing your virtual disk 191. Accessing 191 at this time would cause an error message to be issued because 191 is not yet initialized. Therefore, virtual disk 191 cannot be used.

```
R;
```

Next, enter:

```
   q v stor
```

3M of storage is required to load CMS using the CMS minidisk address. Check virtual storage to confirm that you have at least 3M. If not, you need to issue **def stor 3m** and reIPL 190.

```
   terminal mode vm
```

Allows CMS commands to be entered.

```
   format 191 a

FORMAT WILL ERASE ALL FILES ON DISK 'A(191)'.
DO YOU WISH TO CONTINUE? (YES|NO):

   yes

ENTER DISK LABEL:

   xxxxxx
```

Enter the one-to six-character alphameric label of the virtual disk. You can use whatever label you wish for this virtual disk.

```
FORMATTING DISK 'A'.

'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

'nn' is the number of cylinders assigned to 191 in the directory.

- If you do not have a second terminal available, logoff. Then, log back on using the MAINT id.

  logoff Enter only if no second terminal available.

  logon maint cpcms

- If you have a second terminal available:

  – At the operator console:

    enable all

  – At the 2nd terminal:

    logon maint cpcms

Proceed to Step 14.

Step 14. Updating the SYSTEM NETID File for CMS Productivity Aids (Optional)

The SYSTEM NETID file is referenced when you issue CMS commands to communicate via the network. The CMS IDENTIFY command is used by the NOTE, SENDFILE, and TELL commands to transmit notes, files, and messages.

The IDENTIFY command uses the CP QUERY CPUID command to retrieve the CPU serial number. This number is then looked up in the SYSTEM NETID file. IDENTIFY uses the CP QUERY USERID command to retrieve the node. If there is a conflict in nodes between the SYSTEM NETID file and CP QUERY USERID, the node in SYSTEM NETID takes precedence. If there is no record with a matching serial number, or if the SYSTEM NETID file is not found, the 'rscsid' is set to an asterisk (*).

The logon of the MAINT userid causes an automatic IPL 190.

```
VM/SP REL n  MM/DD/YY HH:MM

  ENTER
```

Completes the automatic IPL.

```
R;

  set emsg on
```

Provides message numbers for messages displayed in this step.

```
R;

  acc 190 b

R;
```

Running from the MAINT userid allows you to write to the 190 minidisk where the SYSTEM NETID file will be stored.

If you have an RSCS Networking virtual machine, you should update a file on the CMS 190 disk called 'SYSTEM NETID'. If you plan to use the TELL, NOTE, or SENDFILE command, you should update the supplied file. The records in this file have the format:

```
ssssss nodeid rscsid
```

where 'ssssss' is your CPU serial number, 'nodeid' is your local node, and 'rscsid' is the userid of the RSCS virtual machine.

To obtain your CPU serial number, issue:

```
  q cpuid

CPUID = FFssssssdddd0000

R;
```

'ssssss' is your CPU serial number and 'dddd' is your processor device type.

```
  xedit system netid b2

  file

R;
```

Update the SYSTEM NETID file with your CPU serial number (ssssss), local node (nodeid), and userid of the RSCS virtual machine (rscsid).

Proceed to Step 15.

Step 15. Save CMS

By using the DMKSNT layout and the new CP nucleus, you can now save your CMS system.

The default virtual storage size when running under the MAINT userid is 6M. That is the required storage for this step.

Note: In the procedure that follows, be sure to enter the SAVESYS CMS command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMS system, enter:

```
  ipl 190 clear
VM/SP REL n  MM/DD/YY HH:MM
```

The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

```
  savesys cms
```

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide*.

To complete the IPL process:

```
 ENTER
SYSTEM SAVED
VM/SP RELEASE n  MM/DD/YY HH:MM
 ENTER
```

Your CMS system is now saved. You can issue IPL CMS instead of IPL 190, when you wish to run CMS.

Step 16. Create a Backup Tape (Optional)

This step is optional if the product will not be installed or if segments will not be saved.

The DASD Dump Restore (DDR) program is described in the *VM/SP Operator's Guide.* If your system residence volume is labeled VMSRES at virtual address 123 and you have attached a scratch tape to MAINT at 181, use the following DDR control statements to back it up.

> **Note:** If you did not apply service to your new VM/SP system and did not load the source files, you only have to backup your VMSRES; however, if you have done one of those things, you may have to do more work:
>
> - If you applied service to your Release 3.0 system and you installed it on 3310s, 3330s, or 3340s, you also need to create a backup of your VMPK01 volume because that is where the minidisks for MAINT 293 and 294 are located.
> - All users who loaded the CP and CMS source files will have to backup their VMSTGE volume, while users who installed their system on 3310s or 3340s will also have to create a backup of their second source volume VMSTG2.
>
> This assumes that the default minidisk locations are not changed.

```
   ddr

ENTER:

   input 123 3330 vmsres

ENTER:

   output 181 3420 (mode 6250

ENTER:

   dump cpvol
```

dump cpvol causes cylinder 0 and any disk cylinders allocated as PERM or DRCT to be dumped onto the tape.

```
DMKDDR711R VOLID READ IS VMSRES
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:

   yes

ENTER NEXT EXTENT OR NULL LINE

   ENTER

DUMPING   VMSRES
END OF DUMP

   ENTER

END OF JOB
PRT FILE nnnn  FOR VMSP3    COPY 001      HOLD
R;
```

You have completed the VM/SP Installation procedure. Verify that your new CP and CMS are working properly by turning to page 101 for the Installation Verification Procedure.

Chapter 3. Installation When Not Using the Starter System Tape

This chapter applies to users who have a prior release of VM/SP and wish to install a new VM/SP system using the merged Product Tape. The procedure in this chapter accomplishes the following objectives:

- Prepare DASD space
- Load the Product Tape
- Generate a new level CMS system
- Perform optional PREP EXEC tasks
- Apply service updates
- Alter the directory and DMKxxx files
- Generate a new CP nucleus
- Create a SYSTEM NETID file (optional)
- Save CMS
- Create a backup tape (optional).

Note: The procedure documented in this chapter describes only one of many ways to update your current system.

Warning: Before proceeding with the installation of VM/SP without the Starter System, use the DDR program to create a backup of the current system. You can recover any portion of your current system from this DDR tape if necessary. See *VM/SP Operator's Guide* **for assistance.**

The installation tools, GENERATE EXEC and PREP EXEC, contain new release functions for some options. It is necessary to run under the new VM/SP CMS system as soon as possible to avoid module incompatibility. This procedure loads the Product Tape and builds the CMS nucleus as the first step of a migration process.

Assumptions

- The person doing this generation has some VM/SP system programming experience. Therefore, the steps for this procedure are not as detailed as those for the Starter System procedure described in "Chapter 2. Installation When Using the Starter System Tape" on page 15.

- All new or modified minidisks are defined in the virtual machine directory for the MAINT userid.

- **A backup copy of your current system** and a print out of the current system directory file are readily available.

- Virtual storage has been defined at three megabytes.

Planning and Preparation

Because some of the requirements for VM/SP Release 3.1 are different from earlier versions of VM/SP, you will have to make some changes before you can install the VM/SP Release 3.1 Merged Product Tape. These changes mainly concern:

- Minidisk sizes
- New minidisks
- DMKSNT entries for saved systems.

Minidisk Sizes

You should carefully compare the size of your current MAINT minidisks with the default sizes from the VM/SP Release 3.1 directory. The default sizes are defined in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

Minidisks concerned are: MAINT 194, 293, 294, 393, and 394. Because MAINT 293 and 294 will be used only when you are applying service to VM/SP, you may chose to change these after the installation process; however, it is advisable to erase their contents at this time to ensure that you will not apply old VM/SP updates to your new VM/SP Release 3.1 system.

DMKSNT Entries for Saved Systems

Three changes are important for Release 3.1:

- CMS and CMSL segments have increased in size
- CMSSEG has been removed, since its coding has been incorporated into the CMS Nucleus
- The load address of INSTVSAM has been changed from FC0000 to FA0000. (Be sure to update your DMKSNT to reflect this address change.)

Because there is not enough space in your current $SAVESYS$ area for the larger CMS and CMSL segments, you should find some free PERM space for them. Be sure to update the DIRECTORY entry for user $SAVESYS$ accordingly.

Procedure Overview

In this procedure you:

- Define a 490 minidisk to replace the current 190 minidisk.
- Load the new CMS files onto the 490 minidisk.
- Load other product files onto respective minidisks.
- Build a new level CMS system and use it for the remainder of the system generation procedure.
- Change the DMKSNT definition (CMS NAMESYS) to reflect the new CMS minidisk placement.
- Generate a new CP nucleus.
- Save and backup the new system.

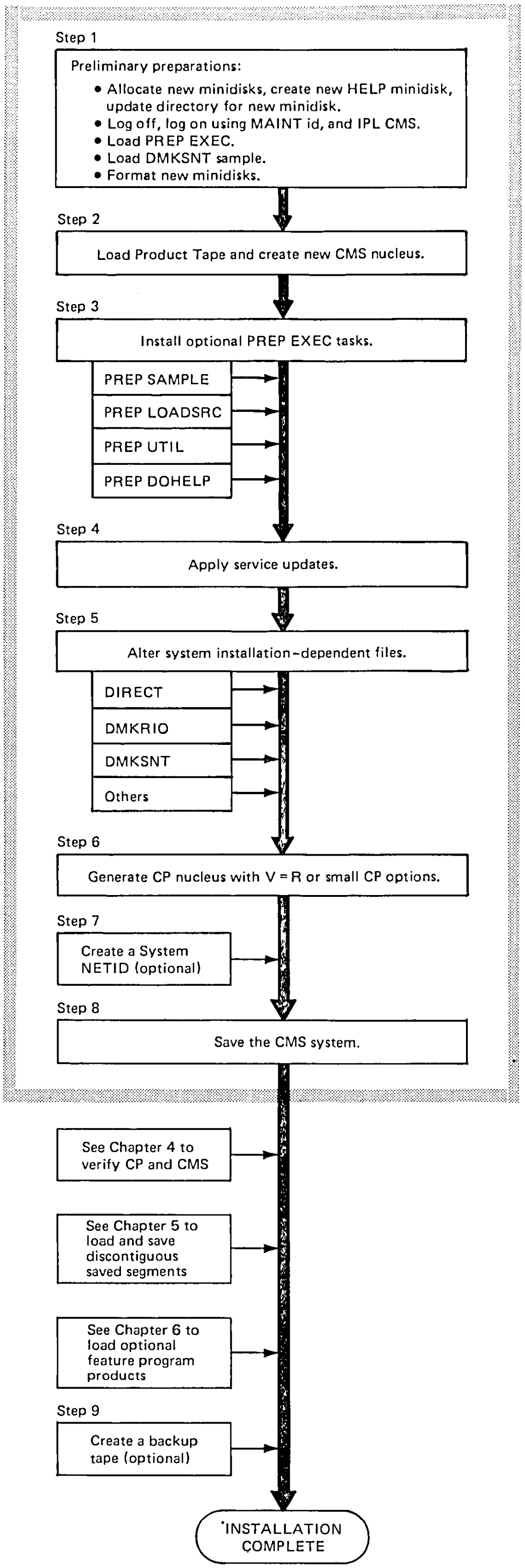

Step 1
Preliminary preparations:
• Allocate new minidisks, create new HELP minidisk, update directory for new minidisk.
• Log off, log on using MAINT id, and IPL CMS.
• Load PREP EXEC.
• Load DMKSNT sample.
• Format new minidisks.
Step 2
Load Product Tape and create new CMS nucleus.
Step 3
Install optional PREP EXEC tasks.
PREP SAMPLE
PREP LOADSRC
PREP UTIL
PREP DOHELP
Step 4
Apply service updates.
Step 5
Alter system installation-dependent files.
DIRECT
DMKRIO
DMKSNT
Others
Step 6
Generate CP nucleus with V = R or small CP options.
Step 7
Create a System NETID (optional)
Step 8
Save the CMS system.
See Chapter 4 to verify CP and CMS
See Chapter 5 to load and save discontiguous saved segments
See Chapter 6 to load optional feature program products
Step 9
Create a backup tape (optional)
INSTALLATION COMPLETE

Figure 12. Using the Merged Product Tape Without a Starter System

Step 1. Preliminary Preparations

1a. Edit your existing system directory

For example:

```
xedit vmusers direct
```

You may want to change the current virtual machine directory[11] for the MAINT userid. If so, consider the following:

- Define a minidisk that will eventually replace the current 190 minidisk. The example in this chapter uses 490 as this minidisk designation. If you want to follow this example, you may need to add MAINT 490 to your directory and issue the DIRECT command. The 490 minidisk may need to be formatted also. Be sure to examine the MAINT 190 and 194 entries in the Sample Directory for your device type provided in:

  Appendix J, "3310 Based System Sample Files" on page 303
  Appendix K, "3330 Based System Sample Files" on page 333
  Appendix L, "3340 Based System Sample Files" on page 363
  Appendix M, "3350 Based System Sample Files" on page 395
  Appendix N, "3370 Based System Sample Files" on page 425
  Appendix O, "3375 Based System Sample Files" on page 455
  Appendix P, "3380 Based System Sample Files" on page 487

  Since CMS has grown, you need to increase the recomp area accordingly. See the table on page 82 for the CMS nucleus size requirements for your device type.

  Notes:

  1. Keeping the 490 minidisk on the same physical volume as the 190 minidisk at the beginning of this procedure will simplify things later.

  2. The PREP EXEC is designed to run from the 191 minidisk accessed as C.

  3. EREP is now packaged separately. No space provision is made on the 190 minidisk for EREP code. A sample directory entry for minidisk MAINT 201 has been provided to contain this code. If you want to follow this example, you may need to add MAINT 201 to your directory and issue the DIRECT command. The 201 minidisk may need to be formatted also.

- Current VM/SP Release 1 users only:

  – HELP files should be put on the default 19D minidisk. (1K blocking is recommended since these files are small.)

[11] See

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

for sample minidisk size definitions.

- A LINK to the disk containing system HELP files is required for each user. Two methods of linking to the system HELP disk follow:

  Place "LINK MAINT 19D 19D RR" in each user's directory entry that contains the virtual address of the HELP files disk.

  or

  Have each user add "LINK MAINT 19D 19D RR" for the HELP disk to their PROFILE EXEC.

1b. Check other current minidisk allocation sizes

Define new minidisks sizes if required.

1c. File the updated VMUSERS DIRECT

1d. Update the current system directory using the DIRECT command

For example:

```
  direct vmusers
```

1e. At this point, you must log off then log on using the MAINT userid, and IPL CMS to be able to access the newly defined minidisks.

Note: The virtual machine size for the MAINT machine is assumed to be 6M (3M minimum).

1f. Load PREP EXEC from the Product Tape (file 1)

To load the first file on the Product Tape to your 191 minidisk, issue:

```
rew 181
acc 191 a       (PREP EXEC must run from the 191 minidisk.)
vmfplc2 load
```

1g. Load DMKSNT 'dasdtype' file from the Product Tape (file 2)

To load the second file on the Product Tape to your 191 minidisk, issue:

```
  vmfplc2 load dmksnt (dasdtype)
```

Use the System Product Editor to change the CMS NAMESYS parameters to match the 490 placement.

- If your newly defined 490 minidisk resides on the same physical volume as the 190 minidisk, then SYSCYL is the only parameter you need to change in the CMS NAMESYS macro.

- If you allocated the 490 minidisk on a different physical volume, then you will need to make appropriate changes to both SYSCYL and VSYSRES (VSYSRES must point to the new volume).

Use the following command:

```
  xedit dmksnt assemble
```

Then, use the PUT and GET commands to get entries from the sample DMKSNT device type file and put them into your current DMKSNT ASSEMBLE file.

Note: An alternative is to use the appropriate sample DMKSNT described in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

as a guide for modifying your current DMKSNT file.

1h. Format all minidisks to be used by the new level product

To format the new 490 minidisk, issue:

```
format 490 b
```

Other minidisks you may want to format include 194, 293, and 294. Remember: Any files you want to save must be moved before formatting a minidisk. To do this use: PREP CMSFMT NOSAMP ONLY.

1i. Format the 490 minidisk with the RECOMP option

The 'xxxxx' values shown below for the FORMAT command are valid if the CMS minidisk sizes listed below are used. A minimum of 97 pages is required for the CMS nucleus. To format the 490 minidisk again, issue:

```
format 490 b xxxxx (recomp
```

where the value of xxxxx is the recomp value:

Device	CMS	Nuc	Recomp	xxxxx	
Type	Allocation	Req	Value	(CMS)	(CMSL)
FB-512	45568 blk	1024	43520	43520	44544
3330	100 cyl	3	94	94	97
3340	236 cyl	7	222	222	229
3350	49 cyl	2	45	45	47
3375	74 cyl	2	70	70	72
3380	45 cyl	1	43	43	44

Note: If you try to use the current CMS minidisk and there is insufficient free space, the RECOMP operand causes an error message to be issued. For this reason, it is recommended that you use the alternate CMS (490) minidisk method.

Continue with Step 2.

Step 2. Load the Product Tape and Build a New CMS Nucleus.

To load the Product Tape, issue:

```
prep load define nuc
```

This results in prompts for addresses of minidisks that will receive the new product code. A CMS nucleus is then created and loaded at the new product level.

Note: Much of the manual generation that was required for installation in Release 1 has been eliminated by the PREP EXEC. PREP EXEC automates the steps previously needed to generate VM/SP. For further insight into exactly what PREP EXEC does, print a copy of it and follow the generation procedure in more detail.

```
' LOAD ' PROCESS HAS BEEN REQUESTED:

'LOAD' PROCESS STARTING.
NO RDR FILES.
YOU HAVE CHOSEN THE OPTION TO BUILD A
CMS NUCLEUS AFTER LOAD.
CAUTION: THIS EXEC PURGES CLASS I READER
FILES.  CONTINUE? (Y OR N)

  y

CAUTION: NO ERROR CHECKING IS
DONE FOR MINIDISK AVAILABILITY,
THE MDISKS TO BE ENTERED IN THE
FOLLOWING RESPONSES MUST BE CURRENTLY
ATTACHED TO YOUR VIRTUAL MACHINE.

SINCE YOU DO NOT WISH TO USE THE
DEFAULT MINIDISKS TO LOAD THE PRODUCT
TAPE, RESPOND TO THE FOLLOWING PROMPTS.

ENTER WHERE CP FILES ARE TO
BE LOADED ( CUU | FSF )
```

FSF = Forward Space File

```
  194
```

194 is the address of the CP text minidisk.

```
ENTER WHERE HELP FILES ARE TO
BE LOADED ( CUU | FSF )
```

19d

19d is the address of HELP minidisk.

```
ENTER WHERE CMS FILES ARE TO
BE LOADED ( CUU | FSF )
```

490

```
THE ATTACHED MDISKS ARE DEFINED AS FOLLOWS:
  CP MDISK = 194
  HELP MDISK = 19D
  CMS MDISK = 490
ARE THESE RESPONSES CORRECT? ( Y | N | EXIT )
```

y

```
' LOAD ' PROCESS STARTING . . .

LOADING   MAINT 194 MINIDISK (CP TEXT DECK DISK)
MAINT 194  LOADED

LOADING   MAINT 19D MINIDISK (HELP FILE DISK)
MAINT 19D  LOADED

LOADING MAINT 490 MINIDISK (CMS TEXT FILE DISK)
MAINT 490 LOADED

THE PRODUCT TAPE HAS BEEN LOADED TO DISK. IF SOURCE IS
NOT TO BE LOADED TO DISK, YOU MAY WISH TO REPLACE
THE PRODUCT TAPE WITH A SCRATCH TAPE AT THIS POINT.
'LOAD' EXECUTION COMPLETE.

GENERATING THE SYSTEM ASSEMBLER COMMAND.
NO RESPONSE IS REQUIRED FOR THE ASSEMBLE PROMPTS.
ASSEMBLE XF GEND PROC
ENTER TARGET DISK MODE FOR ASSEMBLE MODULES.
DEFAULTS TO S-DISK IF NONE ENTERED.
ASSEMBLE XF GEND COMPLETE.
SYSTEM LOAD DECK COMPLETE.
DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

490

The response to this prompt is the address of the device where the CMS nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =
```

94

The response to this prompt is device dependent. See page 82 for the correct value for your device type.

```
DMSINI610R ALSO IPL CYL/BLK 0 ?
```

yes

```
DMSINI611R VERSION IDENTIFICATION =
```

ENTER

You are permitted to enter up to 32 descriptive characters.

```
DMSINI612R INSTALLATION HEADING =
```

ENTER

You are permitted to enter up to 64 descriptive characters.

```
VM/SP REL n MM/DD/YY
DMSACC723I Y (19E) R/O[12]
R;
```

Warning: Do not issue "SAVESYS CMS" at this point.

In order to use the "new" CMS, define the 190 minidisk as 590. Then, define the 490 minidisk (the "new" CMS) as 190 in your virtual machine.

def 190 590

Redefines the "old" CMS to 590.

def 490 190

Redefines the "new" CMS to 190. Necessary so that the "new" CMS can be used in place of the "old" CMS.

IPL the new CMS by issuing:

ipl 190

You *must* IPL by using the "new" CMS minidisk address since you do not wish to load the "old" CMS system.

To verify that the new system is operational, execute some basic CP and CMS commands, for example: QUERY DISK and QUERY DASD.

> **Note:** You can use the GENERATE MAP function at this time to load the map.

Continue with Step 3.

[12] This message is displayed only when the Y-STAT is too large to fit in the CMS nucleus. If desired, the CMS nucleus can be generated to allow space for a large Y-STAT. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for additional information.

Step 3. Perform Optional PREP EXEC Tasks

During the installation process you might want to alter your existing files or tailor some sample files shipped with the system. You may also want to load system source code or load utility programs. There are PREP EXEC options to assist you. They are:

A. PREP SAMPLE - Print out copies of sample files.
B. PREP LOADSRC - Load CP and/or CMS source code.
C. PREP UTIL - Create stand-alone utility program tape.
D. PREP DOHELP - Erase HELP files or convert them to uppercase.

To initiate any PREP EXEC task, enter the PREP command with the appropriate operand, for example: PREP UTIL. If you need more information about a specific command, you can enter the HELP option following the operand. For example, if you want to know more about PREP UTIL, you enter PREP UTIL HELP.

For a more detailed sample, see pages 48 through 51 in "Chapter 2. Installation When Using the Starter System Tape" on page 15.

Continue with Step 4.

Step 4. Apply Service Updates

Note: Any PUT tape *Memo to Users* information replaces or supplements information in this step. Be sure to review it before applying service.

A VM/SP service tape may be supplied with VM/SP. This tape contains cumulative service for VM/SP.

Mount and attach the service tape (with no ring) as virtual drive 181. Now issue the following commands:

```
attach cuu * 181
```

cuu is the real address of the service tape device.

```
access 191 c

vmfplc2 load * * c

vmserv
```

The VMSERV EXEC maps the service tape and allows you to print the PUT document and *Memo to Users*. Reply yes to the next prompt.

```
DO YOU WANT TO PRINT THE MEMO TO USERS?

    yes
```

VMSERV then prints the PUT document and *Memo to Users*, and reminds you to read them prior to installing service. After reviewing the documentation and contacting IBM concerning the latest service activity, you can install service as described in the PUT document and *Memo to Users*.

Do *not* build a new CP nucleus as described in the user memo, but continue here after applying service from the PUT. After applying service, recreate the ASSEMBLE module as follows:

```
access 190 c

asmgend

ipl 190 parm autocr
```

Continue with Step 5.

Step 5. Modify System Installation-Dependent Files

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied with VM/SP serve as a base for building files unique to your installation. You may need to modify one or more of these files now.

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied in:

> Appendix J, "3310 Based System Sample Files" on page 303
> Appendix K, "3330 Based System Sample Files" on page 333
> Appendix L, "3340 Based System Sample Files" on page 363
> Appendix M, "3350 Based System Sample Files" on page 395
> Appendix N, "3370 Based System Sample Files" on page 425
> Appendix O, "3375 Based System Sample Files" on page 455
> Appendix P, "3380 Based System Sample Files" on page 487

are intended to serve as a base for building files unique to your installation. Examine the listings of the Sample Directory and DMK*xxx* files and use the System Product Editor to modify these sample files if desired. For more information, see "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference.*

Directory Considerations

If you are ready to build the CP nucleus, your directory will need to be modified now, reflecting the changes you defined in Step 2. Namely, you need to change MAINT's 190 MDISK to 590 and MAINT's 490 MDISK to 190 before proceeding. You can update this file later to include additional virtual machine definitions and userids. See "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference* for a general discussion. For more detailed information, see "A Virtual Machine for Updating VM/SP" on page 142.

DMKRIO Considerations

Your current DMKRIO file probably has not changed unless you have altered I/O devices with this installation. Make any necessary changes to the DMKRIO files at this point before you build the new CP nucleus.

DMKSNT Considerations

Because CMS and CMSL segments have been modified, you may want to merge the new sample DMKSNT with your existing file. Check user-selected product installation documentation for system name table related information.

A description of CMS saved segment relocation can be found in "Planning for CMS" in Part 1 of the *VM/SP Planning Guide and Reference.* For the procedure regarding alternate CMS nucleus placement, see Appendix E, "Example of Alternate CMS Nucleus Placement" on page 275. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for an example to modify the CMS nucleus to include file status tables for the S- and Y-minidisks.

DMKSYS Considerations

No space dedicated to paging has been provided in the system samples. An additional volume is required for paging space. The new volume id should be added to the SYSOWN macro definition to contain the paging space. Refer to "Direct Access Storage Requirements for CP" in the *VM/SP Planning Guide and Reference* for more information.

During execution of the GENERATE EXEC process, you will be given the opportunity to write the CP nucleus to tape. Be aware that the SYSRES=123 definition in the sample DMKSYS file requires a real address of 123, if this was to be used as a standalone backup tape. If you wish to change this to a real address, you must also change the corresponding DIRECT MDISK 123 statement for the MAINT userid and the DIRECTORY 123 statement in the sample directory.

Other Considerations

The following files can be altered as required to meet your installation's needs.

- Printer Universal Character Set[13]

  DMKUCS (1403)
  DMKUCB (3211)
  DMKUCC (3203)
  DMKPIB (3262)

- Font Offset Buffer[13]

  DMKPIA (3289)

- Forms Control Macros[13]

  DMKFCB (forms control)

- LOGO

  DMKBOX

Continue with Step 6.

[13] See "Updating Printer Modules" on page 192 for further information.

Step 6. Generating a CP Nucleus

The procedure for generating a CP nucleus is managed by GENERATE EXEC. GENERATE EXEC provides entry points for generating selected portions of the CP nucleus. Figure 10 on page 54 shows the entry points and exits of the sequence.

An entry point is selected by specifying the appropriate operand following the GENERATE command. For instance, placing the TEMP operand after GENERATE causes a complete CP nucleus to be created.

Notes:

1. **You must reassemble all DMKRIO, DMKSYS, and DMKSNT modules before generating the CP nucleus.**
2. If you have system TEXT files like DMKRIO, DMKBOX, DMKSYS, DMKSNT, etc. on you 191 disk, you should rename or erase them to avoid having duplicate files, because the GENERATE EXEC will access your 194 disk as A and put the system TEXT files on the 194 disk if there is not a GENERATE DEFAULTS file.

Four of the operands allow an ONLY option. If "ONLY" is specified after the DIRECT, DMKRIO, DMKSYS, or DMKSNT operand, *only* the GENERATE code associated with the specific operand is executed.

If you would like to display HELP information that describes any GENERATE entry point, enter:

generate help

For more detailed information on the GENERATE EXEC, see "GENERATE" on page 197.

If you are using one tape drive only, issue:

def 181 182 Allows you the option to save the CP nucleus on tape later in this step.

Replace the tape currently mounted with a scratch tape (with ring) to receive the nucleus code.

If you plan to include the Small CP or Virtual=Real option, read Appendix G, "Special Options for CP" on page 283 before proceeding.

This procedure assumes that you enter GENERATE EXEC from the top. To start, get temporary disk space by entering:

generate temp

The TEMP option causes execution of GENERATE EXEC from the top through generation of a CP nucleus. The GENERATE TEMP option formats and allocates temporary minidisk 192 (B-disk) for assembler workspace. If an error occurs, the routine exits at the step being processed.

```
GET TEMP SPACE ON DASD TYPE = 3330
DISK 'B' NOT ACCESSED
DEV 192 DOES NOT EXIST
DASD 192 DEFINED
FORMAT - TEMP 30 CYLINDER MDISK.
TEMP MDISK 192 ACCESSED AS FILE MODE B.
```

Executing GENERATE TEMP code.

```
DO YOU WISH TO UPDATE THE SYSTEM
DIRECTORY AND ASSEMBLE THE SYSTEM
DEFINITION FILES NOW?
RESPOND ( YES | NO )
```

yes

no results in a temporary minidisk being allocated.

If your directory file is not on one of MAINT's minidisks, reply 'SKIP' to the request to ENTER DIRECTORY FILENAME.

If your directory file is on one of MAINT's minidisks, enter the correct directory name or 'SKIP'. If you still have an old VMUSER DIRECT file somewhere, you could be left without a running system if the old VMUSER DIRECT file is brought on-line.

```
'194' REPLACES ' A (191) '
C (191) R/O
ASSEMBLE FILE SEARCH ORDER IS 194 THEN 191.
```

Executing GENERATE VMSP code.

```
ENTER DIRECTORY FILENAME:
THE NULL DEFAULT VALUE IS 'VMUSERS'.
TO BYPASS STEP ENTER 'SKIP' -
(GO TO DMKRIO ASSEMBLE STEP)
```

Executing GENERATE DIRECT code.

ENTER

```
EOJ DIRECTORY UPDATED
```

The following response will not appear if a GENERATE DEFAULTS file has been generated.

VMUSERS is now your directory filename.

```
WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:    ( NO | YES )
```

no

This sample procedure assumes a uniprocessor system. Executing GENERATE DMKRIO code.

```
VMFASM BEING EXECUTED WITH
FN= DMKRIO CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKRIO
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKRIO TEXT CREATED
```

Executing GENERATE DMKSYS code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSYS CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSYS
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSYS TEXT CREATED
```

Executing GENERATE DMKSNT code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSNT CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSNT
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSNT TEXT A1 CREATED
'194' REPLACES ' A (194) '
```

If any errors occurred while the directory was being built or while the DMKxxx files were assembling, the directory program and the GENERATE EXEC procedure issued the message:

```
EDIT AND CORRECT THE DIRECTORY FILE IN ERROR.

                    - or -

EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR.
```

If you received one of these messages, read the next page for the procedure to correct the error condition. Otherwise, continue on page 94.

Read this page if you received one of the error messages on the previous page.

ERROR CONDITIONS

A. "EDIT AND CORRECT THE DIRECTORY FILE IN ERROR" Message

Errors were detected while the Directory was being loaded.

The DIRECT Option reads the directory file from your minidisk and writes it to the system residence volume. Specifying GENERATE DIRECT ONLY causes an exit from the GENERATE routine after the new directory is written. If "ONLY" is not specified, the routine continues with assembly of DMKRIO, DMKSYS, and DMKSNT. You need to edit and correct the Directory file using the System Product Editor. Then, respond with:

generate direct

B. "EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR" Message

Errors were detected during assembly of one or more of the DMKRIO, DMKSYS, or DMKSNT files.

Use the System Product Editor to correct the indicated "filename" (DMKRIO, DMKSYS, or DMKSNT). Then, enter the GENERATE command with the appropriate option. See the following example:

generate dmkxxx

dmkxxx represents **dmkrio, dmksys, or dmksnt.**

This message appears only when generating DMKRIO.

```
WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:    ( NO | YES )
```

no

This sample procedure assumes a uniprocessor system.

```
VMFASM BEING EXECUTED WITH
FN= DMKxxx CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKxxx
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKxxx TEXT CREATED
```

- If **dmkxxx** = **dmkrio**, then GENERATE DMKRIO is executed. The DMKRIO Operand assembles DMKRIO. If the ONLY option is not specified, it also:

  1. Assembles DMKSYS and DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksys**, then GENERATE DMKSYS is executed. The DMKSYS Operand assembles DMKSYS. If the ONLY option is not specified, it also:

  1. Assembles DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksnt**, then GENERATE DMKSNT is executed. The DMKSNT Operand assembles DMKSNT. If the ONLY option is not specified, it also generates a CP nucleus.

The procedure continues on the next page.

The following three responses will not appear if a GENERATE DEFAULTS file has been generated.

```
DO YOU WANT THE SMALL CP OPTION? --
ENTER:  ( NO | YES )
```

See Appendix G, “Special Options for CP” on page 283 if you want the Small CP option.

no

Executing GENERATE CP NUCLEUS. The CP NUCLEUS operand causes the CP nucleus to be generated. A CP nucleus can be written to tape or to the target disk. If the NOLOAD option is specified, the resulting nucleus will not be IPLed automatically.

```
WHICH ARE YOU GENERATING: A
'UNI PROCESSOR' (UP),
AN 'ATTACHED PROCESSOR' (AP),
OR A 'MULTI PROCESSOR' (MP) SYSTEM ?

ENTER: ( UP | AP | MP )
```

Your responses to these questions determines the CNTRL file and LOADLIST EXEC used to correctly build your system.

up

```
VIRTUAL=REAL OPTION REQUIRED (YES|NO):
```

See Appendix G, “Special Options for CP” on page 283 if you want the V=R option.

no

```
READER EMPTY
VMFLOAD BEING EXECUTED
LOADLIST= CPLOAD AND CNTRL=DMKSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn  TO  MAINT    COPY 001   NOHOLD

AN IPL'ABLE CP NUCLEUS EXISTS IN YOUR VIRTUAL CARD READER.
DO YOU WISH TO CREATE AN IPL'ABLE TAPE (FOR BACK-UP),
OR, LOAD THE NUCLEUS IN YOUR READER DIRECTLY
TO YOUR SYSTEM RESIDENCE DISK VOLUME?
ENTER: ( DISK | TAPE | EXIT ).
```

tape

You are saving the new CP nucleus on tape as well as on DASD. If any errors are detected while the tape is being written, you must recreate the CP nucleus. To do this, enter GENERATE CP NUCLEUS. The procedure then restarts at the point above where you are asked: `DO YOU WANT THE SMALL CP OPTION? --`.

```
IPLABLE NUCLEUS NOW ON TAPE ****

WHEN YOU RECEIVE THE MESSAGE: NUCLEUS LOADED ON 'volid'
ENTER: 'IPL 190 PARM AUTOCR'
TO PUT THE CP LOAD MAP ONTO YOUR 194 MINIDISK, AFTER THE IPL
ENTER: GENERATE MAP

NUCLEUS LOADED ON VMSRES - STARTING CYL/BLK=399,
LAST CYL/BLK USED=402
```

The message below does not automatically display on many processors. Check your processor reference manual for information about displaying and interpreting the PSW.

```
CP ENTERED; DISABLED WAIT
PSW '00020000 00000012'
```

This is the normal return if the nucleus loaded correctly.

If you do not receive the **NUCLEUS LOADED ON 'volid'** message, read the following procedure to correct the error condition. Otherwise, continue on page 96.

ERROR CONDITION

No "NUCLEUS LOADED ON 'volid'" Message

- Inspect the error load map in the virtual reader and the virtual PSW. A loader error may be indicated on the listing or PSW. See the *VM/SP System Messages and Codes* for a list of the loader wait state codes.

  After correcting the error, issue GENERATE CP NUCLEUS.

- If you shut down following a nucleus loading error, you may not be able to reIPL the system. If so:

  1. Mount the tape created by the GENERATE command in the initial generation procedure.
  2. IPL that tape.
  3. Recreate your directory using GENERATE DIRECT ONLY, and generate a new CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete. To reload CMS, enter:

ipl 190 parm autocr

For *second level* operation, ENTER after IPLing.

```
VM/SP n STARTER
R;
```

generate map

```
LOAD MAP HAS BEEN PLACED IN READER
```

Executing GENERATE MAP to load the CP nucleus map onto the 194 minidisk.

```
SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cp

```
'194' REPLACES ' A (194) '
YOU HAVE NOW SAVED "CPNUC MAP" ON MDISK "194"

R;
```

The contents of the load map are described in "Chapter 9. Updating An Installed VM/SP System" on page 165. Two external names may be listed as undefined on the load map. The external name DMKSLC is undefined if the Virtual=Real option is not specified. The external name DMKRNTBL is undefined if there is no entry in the system name table for a 3704/3705 control program (Provided you did not code a NAMENCP macro in the DMKSNT file). Also, other names might be listed as undefined if other modules were deleted as described in "Reducing the CP Nucleus Size" in Part 1 of the *VM/SP Planning Guide and Reference.*

You have now generated the CP nucleus having had the chance to select Small CP or V=R options. At this point continue with Step 7.

Step 7. Updating the SYSTEM NETID File for CMS Productivity Aids (Optional)

The SYSTEM NETID file is referenced when you issue CMS commands to communicate via the network. The CMS IDENTIFY command is used by the NOTE, SENDFILE, and TELL commands to transmit notes, files, and messages.

The IDENTIFY command uses the CP QUERY CPUID command to retrieve the CPU serial number. This number is then looked up in the SYSTEM NETID file. IDENTIFY uses the CP QUERY USERID command to retrieve the node. If there is a conflict in nodes between the SYSTEM NETID file and CP QUERY USERID, the node in SYSTEM NETID takes precedence. If there is no record with a matching serial number, or if the SYSTEM NETID file is not found, the 'rscsid' is set to an asterisk (*).

The logon of the MAINT userid causes an automatic IPL 190.

```
VM/SP REL n  MM/DD/YY HH:MM

   ENTER
```

Completes the automatic IPL.

```
R;

   set emsg on
```

Provides message numbers for messages displayed in this step.

```
R;

   acc 190 b
```

Running from the MAINT userid allows you to write to the 190 minidisk where the SYSTEM NETID file will be stored.

```
R;
```

If you have an RSCS Networking virtual machine, you should update the file on the CMS 190 disk called 'SYSTEM NETID'. (If you plan to use the TELL, NOTE, or SENDFILE command, you should create this file.) The records in this file have the format:

```
ssssss nodeid rscsid
```

where 'ssssss' is your CPU serial number, 'nodeid' is your local node, and 'rscsid' is the userid of the RSCS virtual machine.

To obtain your CPU serial number, issue:

```
   q cpuid

CPUID = FFssssssdddd0000
```

'ssssss' is your CPU serial number and 'dddd' is your processor device type.

```
   xedit system netid b2

   file
```

Update the SYSTEM NETID file with your CPU serial number (ssssss), local node (nodeid), and userid of the RSCS virtual machine (rscsid).

```
R;
```

Proceed to Step 8.

Step 8. Save CMS

By using the DMKSNT layout and the new CP nucleus, you can now save your CMS system.

The default virtual storage size when running under the MAINT userid is 6M. That is the required storage for this step.

Note: In the procedure that follows, be sure to enter the SAVESYS CMS command before **ENTER**. (A null entry usually causes IPL execution to complete.)

To save the CMS system, enter:

ipl 190 clear

```
VM/SP REL n   MM/DD/YY HH:MM
```

The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

savesys cms

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP RELEASE n   MM/DD/YY HH:MM
```

Your CMS system is now saved. You can issue IPL CMS instead of IPL 190, when you wish to run CMS.

Note: EXEC procedures under CMS that help you load, set storage keys for, and save CMS discontiguous saved segments are:

EXEC	**DSS Named Segment**
DOSGEN	CMSDOS and INSTVSAM
VSAMGEN	CMSVSAM and/or CMSAMS
SAMGEN	CMSBAM
GENERATE	CMSL

The information you need to complete this task is contained in "Chapter 5. Loading and Saving Discontiguous Saved Segments" on page 109. Other essential material is located in "Generating Saved Systems" in the *VM/SP System Programmer's Guide.*

Step 9. Create a Backup Tape (Optional)

The DASD Dump Restore (DDR) program is described in the *VM/SP Operator's Guide*. If your system residence volume is labeled VMSRES at virtual address 123 and you have attached a scratch tape to MAINT at 181, use the following DDR control statements to back it up:

```
   ddr

ENTER:

   input 123 3330 vmsres

ENTER:

   output 181 3420 (mode 6250

ENTER:

   dump cpvol
```

dump cpvol causes cylinder 0 and any disk cylinders allocated as PERM or DRCT to be dumped onto the tape.

```
DMKDDR711R VOLID READ IS VMSRES
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:

   yes

ENTER NEXT EXTENT OR NULL LINE

   ENTER

DUMPING   VMSRES
END OF DUMP

   ENTER

END OF JOB
PRT FILE nnnn  FOR VMSP3    COPY 001     HOLD
R;
```

You have completed the VM/SP Installation procedure. If you wish to verify that your new CP and CMS are working properly, turn to page 101 for the Installation Verification Procedure.

Chapter 4. Verifying CP and CMS Using the IVP (Optional)

The Installation Verification Procedure (IVP) for VM/SP is optional. It checks that CP and CMS are working properly. The two files that make up the IVP use the EXEC facility of CMS. Requirements for the fully functional IVP include:

- Two independent virtual machines
- At least one terminal
- The system operator's virtual machine and console.

The tests verify the following CP areas:

- Automatic warm start after abnormal termination of VM/SP
- Correct EC level for machines with ECPS/VSE
- Message transmission to the system operator
- Spooled data transfer to other virtual machines
- I/O spooling
- Disk I/O support
- Off-line I/O operations
- Paging operations
- Task dispatching and scheduling
- Multiple virtual machine support.

The tests verify the following CMS facilities:

- Creating and modifying files via XEDIT command
- Copying files
- Printing and punching CMS files
- Formatting disks
- Assembling executable programs
- Executing user programs
- Stacking and unstacking command and data input from the terminal
- Using multilevel nested EXEC procedures
- Communicating with the user from EXEC procedures
- Creating and executing user-written commands
- Processing normal CMS commands
- Issuing commands to CP.

Several other system facilities, apart from the primary IVP tests, are checked. The IVP *does not* verify system facilities such as preferred execution options, virtual=real, OS ISAM, and Remote Spooling Communications Subsystem (RSCS). The IVP requires operator intervention when a decision is to be made or when starting the IVP tests themselves. All file creation, erasure, management, and logoff of the virtual machines (with the exception of the system operator) at test completion takes place without operator or user action.

The IVP tests use only the system-provided facilities. The IVP creates, assembles, and later erases all unique test programs.

Facilities Required for Each IVP Virtual Machine

All VM/SP configurations are supported. The IVP executes under the control of CMS. The other facilities required are:

- The assembler
- One virtual read/write disk accessed as the A-disk (usually 191)
- 3M of virtual storage (16M for IVPM1).

Starting the IVP

For a formal completion of the initial installation process, the IVP should be executed. Two virtual machines (IVPM1 and IVPM2), which must be described in the VM/SP directory, are recommended.

The directory entries for the IVP virtual machines, IVPM1 and IVPM2, are included in the VM/SP directory supplied with the Starter System; these entries must be included in your own directory if you plan to run the IVP. Also, the spooling classes for the reader and the punch must be the same.

Note: If you plan to run the IVP procedure using a single terminal device, read the section below. Choose a method to use, follow the directions in the section and continue on the next page. Otherwise, just continue at the top of the next page.

Variations of the IVP (Using One Terminal)

If you wish, you may run the IVP procedure after initial installation using any one of the following methods:

A. To execute the IVP without testing system abnormal termination:

- Retain the created virtual machines in your VM/SP directory.

- Execute the IVP as described in the section, "Starting the IVP," but do not select the "system abnormal termination" option.

B. To execute the IVP with virtual machines other than IVP1 and IVP2:

- Enter "ivp 1 userid1" in place of the command "ivp 1."

- Enter"ivp 2 userid2" in place of the command "ivp 2."

  "userid1" and "userid2" identify the two virtual machines in which the EXEC procedures "ivp 1" and "ivp 2" are to be respectively executed.

C. To execute the IVP in a single virtual machine:

- Enter the command "ivp *" from any logged-on virtual machine.

  This causes the IVP tests to be run in that single virtual machine. When you execute the IVP in a single virtual machine, inter-machine functions, such as transferring data between virtual machines, are not verified. Inter-machine transfer of data is simulated by transferring virtual punched output to the same virtual machine's virtual card reader.

From the system operator's console, first enter:

```
ipl cms
```

Than initiate the IVP tests by entering:

```
ivp
```

Next, answer the following question:

```
ARE YOU THE SYSTEM OPERATOR?
ENTER ''YES'' OR ''NO'':

  yes (or no)
```

A **no** reply defaults to the single virtual machine verification procedure.

Prompting instructions are displayed whenever you must perform an operation or issue a command.

From this point use a second terminal device (one other than the operator's console device).

Log on the virtual machine (IVPM1), using the password IVPASS, and IPL CMS to continue the testing procedure:

```
  logon ivpm1
ENTER PASSWORD:
  ivpass
LOGON AT hh:mm:ss EST FRIDAY mm/dd/yy
  def stor 16m
STORAGE=16384K
```

IPL CMS by issuing:

```
  ipl 190
```

The system will respond:

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

If the IVPM1 191 minidisk is not already formatted, enter:

acc (nodisk

The NODISK option prevents CMS from automatically accessing your virtual disk 191. (Accessing 191 at this time would cause an error message to be issued because 191 is not yet initialized, and therefore cannot be used.)

After the ready message is displayed, issue:

format 191 a

The CMS FORMAT command prompts you with:

```
FORMAT WILL ERASE ALL FILES ON DISK 'A(191)'.
DO YOU WISH TO CONTINUE? (YES|NO):
```

yes

```
ENTER DISK LABEL:
```

ivp1

Enter the one-to-six character alphameric label of the virtual disk. You can use whatever label you wish for this virtual disk.

```
FORMATTING DISK 'A'.
'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

Message issued by CMS.

Then you begin the IVP procedure by issuing:

ivp 1

At this point, the tests begin on virtual machine 1. After the disconnect message is displayed, follow the prompting messages that are displayed. These messages tell you to log on the IVPM2 virtual machine (this may be done on the same terminal), as follows:

logon ivpm2

ivpass

```
LOGON AT hh:mm:ss EST FRIDAY mm/dd/yy
```

IPL CMS by issuing:

ipl 190

The system will respond:

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

If the IVPM2 191 minidisk is not already formatted, enter:

acc (nodisk

The NODISK option prevents CMS from automatically accessing your virtual disk 191. (Accessing 191 at this time would cause an error message to be issued because 191 is not yet initialized, and therefore cannot be used.)

After the ready message is displayed, issue:

format 191 a

The CMS FORMAT command prompts you with:

```
FORMAT WILL ERASE ALL FILES ON DISK 'A(191)'.
DO YOU WISH TO CONTINUE? (YES|NO):
```

yes

```
ENTER DISK LABEL:
```

ivp2

Enter the one- to six-character alphameric label of the virtual disk. You can use whatever label you wish for this virtual disk.

CMS then issues:

```
FORMATTING DISK 'A'.
'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

Then you begin the IVP procedure by issuing:

ivp 2

At this point, the remainder of the tests begin on virtual machine 2. The final phase of the IVP tests consists of displaying, printing, and punching a file that contains the messages generated by IVPM1 after it is disconnected.

Upon successful completion of the tests, the IVP EXEC procedure logs off. If the IVP EXEC does not complete successfully, the system forces an ABEND dump of VM/SP.

Interpreting the Test Results

Messages at the end of the IVP test indicate successful completion. If any errors are detected by the IVP, call IBM for software support bècause an error usually indicates a serious malfunction of the generated system. The IVP procedure identifies each command being tested just before the command is executed.

Error messages are displayed in a four-line format, for example:

```
*** IVP FAILURE HAS OCCURRED ***
*** COMMAND: STATE IVPTST *
*** EXPECTED RETURN CODE 28
*** RECEIVED RETURN CODE 0
```

These messages indicate that the CMS STATE command had a return code of 0 instead of the expected 28.

All information messages that originate within the IVP are preceded by three asterisks (***).

If any command fails, the IVP procedure terminates. Follow the instructions (if any are given) to log off the virtual machine.

Once the IVP procedure has executed successfully, log on your new system as MAINT. Issue:

logon maint cpcms

Then, continue the system installation process by loading and saving discontiguous saved segments, as described in "Chapter 5. Loading and Saving Discontiguous Saved Segments" on page 109.

Chapter 5. Loading and Saving Discontiguous Saved Segments

Information Regarding Discontiguous Saved Segments

The DMKSNT ASSEMBLE file supplied with the Product Tape includes entries for saved segments called CMS, CMSL, CMSVSAM, CMSAMS, CMSDOS, CMSBAM, and INSTVSAM in addition to the saved segments required by the optional feature program products. See "Preparing the System Name Table File (DMKSNT)" in Part 2 of the *VM/SP Planning Guide and Reference* for more information.

Throughout the following discussion, it will be helpful for you to refer to Figure 13 on page 111, which shows where the CMS discontiguous saved segments are loaded in virtual storage using the suggested system name table (DMKSNT) file.

Before a discontiguous saved segment can be attached and detached by name, it must be loaded and saved. It must be loaded at an address that is beyond the highest address of any virtual machine that will attach it. Make sure the saved segment is loaded at an address that does not overlay the defined virtual machine or any other saved segment that may be attached at the same time. Determine the load addresses by checking the entries coded in your DMKSNT module.

> The load address for the discontiguous saved segment should be slightly higher than the highest storage address of the virtual machine using it. If the load address is unnecessarily high, real storage is wasted because CP must have segment table entries for unused storage.
>
> For example, assume you have five CMS virtual machines in your installation. Also assume that all five use the CMS support for VSE program development and testing which is in a 64K segment named CMSDOS. If each of your five CMS virtual machines has a machine size of 320K, you should load the CMSDOS segment just beyond 320K but below 976K (so as to contain it within 1M). Otherwise, real storage would be wasted because CP must maintain segment table entries for each 1024K of storage.

Once the named segment is loaded at the correct address, it can be saved. Make sure that a discontiguous saved segment has storage protection. Set the storage key for the segment with the SETKEY command. The CMS SETKEY command is described in the *VM/SP System Programmer's Guide.*

CMS has EXEC procedures that help you load, set storage keys, and save the CMS discontiguous saved segments. These procedures are as follows:

- DOSGEN EXEC which loads and saves CMSDOS and INSTVSAM segments.
- VSAMGEN EXEC which loads and saves CMSVSAM and/or CMSAMS segments.
- SAMGEN EXEC which loads and saves the CMSBAM segment.
- GENERATE EXEC which loads the CMSL nucleus.

Note: All of the procedures used for loading and saving discontiguous saved segments are filemode number 2. The associated text files are filemode number 1 to reduce the amount of storage needed for the master file directory in the user's virtual machine. The system disk must be accessed as any mode other than S, before loading any discontiguous saved segment. Make sure that this is done *after* any intermediate IPL of CMS.

You may want to compare this storage layout with the suggested DMKSNT layout for your DASD type shown in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487

Note that as new releases of VSE/VSAM become available, they may require more segments, thus requiring all segment addresses above these segments to be increased accordingly.

Note: Refer to the latest VM/SP *Memo to Users* for any changes to the sample layout of CMS storage for discontiguous saved segments.

Relationship of Page Numbers, Segment Numbers, and Hexadecimal Addresses

Since the NAMESYS macro requires you to specify page and segment numbers, and the DOSGEN, SAMGEN, and VSAMGEN procedures require you to enter hexadecimal addresses, you may find the following reference information useful.

1K	=	1024	= X'400'			
4K	=	4096	= X'1000'	=	1 page	
64K	=	65536	= X'10000'	=	16 pages	= 1 segment

To convert a page number to a segment number, divide the page number by 16.

Since one segment is 10000 in hexadecimal, then 20000 is segment 2, 100000 is segment 16, 180000 is segment 24, and so on.

16320K	INSTVSAM	Contains the CMS/DOS discontiguous saved segment used to install CMSVSAM, CMSAMS, and CMSBAM.	FA0000 [1]	4000 [2]	250 [3]
16128K	CMSL	Contains the CMSL nucleus code for the file system, device I/O interrupt handlers, debug, etc., and the 'S' and 'Y' disk directories. This entry is shared.	F00000	3840	240
8192K		END OF DEFINED VIRTUAL STORAGE			
		Contains the CMS control blocks and free storage used during installation of the segment.	350000	848	53
3392K	CMSBAM	Contains VSE routines. The entire segment area is shared.	320000	800	50
3200K	CMSDOS	Contains VSE simulation routines. The area from 3136K to 3200K is shared.	310000	784	49
3136K	CMSAMS	Contains CMS Access Method Services support. The area from 2560K to 2944K is shared; the area from 2944K to 3136K is not.	280000	640	40
2560K	CMSVSAM	Contains CMS VSAM support. The area from 2112K to 2496K is shared; the area from 2496K to 2560K is not.	210000	528	33
2112K		Contains vital free storage pointers.	200000	512	32
2048K	CMS	Contains the CMS nucleus code for the file system, device I/O interrupt handlers, debug, etc., and the 'S' and 'Y' disk directories. This entry area is shared.	190000	400	25
1856K		CMS virtual machine area.	000000	0	0

Note:

[1] Hex load address

[2] Starting page number

[3] Starting segment number

Figure 13. Suggested Layout of Storage for CMS Discontiguous Saved Segments

Procedure for Loading and Saving Discontiguous Saved Segments

The recommended procedure for loading and saving the discontiguous saved segments as displayed in Figure 13 on page 111 follows.

Note: Log on to MAINT if you have not already done so.

Step 1. Load and save INSTVSAM

Use the DOSGEN EXEC procedure to load and save the CMS/DOS segment called INSTVSAM. This CMS/DOS segment is used only for the installation of VSAM (CMSVSAM), Access Method Services (CMSAMS), and VSE simulation routines (CMSBAM).

Before you invoke DOSGEN to load the INSTVSAM segment, IPL CMS in a virtual machine with 16M (16384K) of storage. The INSTVSAM segment will be loaded near the top of storage, at 16128K (hexadecimal address FA0000).

You can increase your virtual machine storage size, up to the maximum size defined for it in the VM/SP directory, by entering the DEFINE STORAGE command. After the DEFINE STORAGE command executes, you must reload CMS. Issue:

def stor 16m	**Note:** The disabled wait state will be entered.
ipl 190	
acc (noprof	

At this point, cancel all global libraries by issuing the commands:

gl maclib	
gl txtlib	
gl doslib	
gl loadlib	
set emsg on	Sets all error messages on.

Release any unneeded disks at this time. Save the CMS/DOS segment called INSTVSAM by issuing:

acc 190 b/a dms* text

```
DISK B(190) R/O
ALSO S DISK
R;
```

dosgen fa0000 instvsam

fa0000 is the virtual storage location, specified with the NAMESYS macro, where the CMS/DOS segment is to be loaded (Figure 13 on page 111). This address is specified in hexadecimal digits.

DOSGEN checks that the address contains valid characters, is greater than X'20000', and less than 16M. If an error is detected, the message

```
DMSGEN095E INVALID ADDRESS 'address'
```

is issued and the command is terminated.

DOSGEN then checks for a read/write A-disk on which to write the CMS loader work file; if none is accessed, it issues an error message and the command terminates.

```
DMSGEN006E NO READ/WRITE A-DISK ACCESSED
```

Next, DOSGEN loads the text files needed for VSE simulation. The text files are loaded starting at the address specified on the DOSGEN command. If there are any unresolved external references, DOSGEN terminates with the message:

```
DMSGEN111E DOSGEN FAILED DUE TO LOAD ERRORS
```

DOSGEN then assigns a storage key of X'D' to the segment and saves it. If an error is detected, one of the following messages is issued and DOSGEN terminates:

```
DMSGEN412S DOSGEN FAILED DUE TO SETKEY ERRORS
DMSGEN141S DOSGEN FAILED DUE TO SAVESYS ERRORS
```

Otherwise, the segment is successfully saved and the load map is written to the A-disk as LOAD MAP A5. The completion message is then issued:

```
SYSTEM SAVED
DMSGEN715I DOSGEN COMPLETE
```

The system name entry in the SYSNAMES table defaults to LOAD.

If you wish to save the load map, access 190 as B and make a copy of the load map now. For example: copy load map a5 instvsam segmap b2.

Step 2. Install CMSBAM

The SAMGEN EXEC procedure is used to load and save the CMSBAM shared segment. The CMSBAM discontiguous saved segment contains the simulated VSE modules necessary to support SAM data management (DTFSD), the ESERV utility program, and VSE/VSAM.

Before invoking the SAMGEN EXEC procedure, you need to define a virtual machine large enough to enable the CMSBAM discontiguous saved segment to be loaded and saved. The size of the virtual machine should be at least 512K greater than the location at which you intend to save CMSBAM.

Note: You must not define your storage larger than the address at which you loaded the INSTVSAM shared segment.

If you are using the example in Figure 13 on page 111, define your virtual machine size as 8192K (8M) by issuing:

```
  def stor 8m
STORAGE=8192K
```

When you redefine storage you will receive the CP ENTERED WAIT STATE message.

Then, issue:

```
  ipl 190
  acc (noprof
  acc 190 g/g
R;
```

Before proceeding, ensure that the following CMS files (distributed as part of VM/SP) are available:

```
CMSBAM DOSLIB
SAMGEN EXEC
```

Issue the following CMS commands:

```
  set sysname cmsdos instvsam
R;
  set sysname cmsbam dummy1
R;
  set dos on
R;
```

dummy1 is any character string that does not correspond to the name of a segment in DMKSNT. If you altered DMKSNT and used a segment name of DUMMY1, you must use a segment name that does not appear in DMKSNT.

Invoke SAMGEN by issuing the following command:

```
samgen
```

You are then prompted to enter the load address for the CMSBAM discontiguous saved segment.

```
SGN363R ENTER LOCATION WHERE CMSBAM WILL BE LOADED AND SAVED
```

```
320000
```

Value from the example in Figure 13 on page 111.

SAMGEN fetches the simulated VSE phases from the CMSBAM DOSLIB. The simulated phases are loaded at the designated address and a storage protection key of X'F' is assigned to the segments. You receive the message:

```
SGN364I FETCHING CMSBAM...
```

This is followed by a message from the FETCH command that describes the entry point of the phase fetched.

SAMGEN then prompts you for the name of the segment to be saved.

```
SGN366R ENTER NAME OF SYSTEM TO BE SAVED
```

To this prompt you should respond:

```
cmsbam
```

SAMGEN saves the segment.

```
SGN365I SYSTEM CMSBAM SAVED
```

Indicates that CMSBAM has been successfully saved. CMSBAM installation is now complete.

Step 3. Install CMSVSAM and Access Method Services (Optional)

Note: If you are NOT installing VSAM, skip Step 3 and continue with Step 4.

If you ARE installing VSAM from an Optional Feature Program Product tape, skip Step 3 and Step 4 and continue with Step 5.

In order to use the CMSVSAM and/or Access Method Services facilities under CMS, the CMSVSAM and CMSAMS discontiguous shared segments must have been saved previously and be available to the virtual machine.

The VSAMGEN EXEC procedure helps you load and save the VSAM and Access Method Services (AMS) segment.

Before you invoke the VSAMGEN EXEC you must:

1. Install the CMS/DOS shared segment INSTVSAM

   - Have the VSAM product tape mounted on 181

   - Access as your A-disk in read/write mode, a minidisk large enough to contain all of the VSAM and AMS text decks required to generate the shared segments. The amount of space required depends on the minidisk device type as shown below:

Device Type	Cylinders/Blocks Required
3330 Model 1 or 11	30
3340	70
3350	15
3375	22
3380	14
FB-512	13440

2. Generate VM/SP with the applicable saved name table and reserve enough DASD space to save all the shared segments.

3. Define a virtual storage size large enough to contain the CMSVSAM and Access Method Services segments, plus the CMSDOS segments, plus at least one segment to contain the CMS control blocks and free storage used during the installation process.

If you follow the example in Figure 13 on page 111, define your virtual storage size as 8192K or 8M.

Note: You must not define your storage larger than the address at which you loaded the INSTVSAM shared segment.

An example of CMSVSAM and CMSAMS shared segment installation follows. The values used for replies to prompts assume the use of the DMKSNT as suggested in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample Files" on page 425
Appendix O, "3375 Based System Sample Files" on page 455
Appendix P, "3380 Based System Sample Files" on page 487
"Preparing the System Name Table File (DMKSNT)" in *VM/SP Planning Guide and Reference*
Figure 13 on page 111.

Ensure that you have the VSAM tape mounted on a tape unit attached to your virtual machine as virtual address 181.

If you have class B authority, issue

```
att xxx * 181
```

Otherwise, ask the system operator to do this for you.

You need to IPL a CMS system with sufficient virtual storage. Issue:

def stor 8m

8m (8192K) comes from the example shown in Figure 13 on page 111.

ipl 190

Access the CMS system disk as any mode B through Z excluding S.

acc (noprof

```
R;
```

acc 190 t/a

```
R;
```

Define the name of the CMS/DOS shared segment you will use to generate the CMSVSAM and CMSAMS segments. Issue:

set sysname cmsdos instvsam

```
R;
```

Set the name of the CMSBAM segment as follows:

set sysname cmsbam nnnn

```
R;
```

nnnn is any character string that does *not* correspond to the name of a segment in DMKSNT.

VSAMGEN can be invoked starting at any of three distinct functions provided certain input requirements are met.

VSAMGEN Function	Required Material
Read the VSAM product tape and create the necessary text files on the A-disk.	VSAM product tape
Link edit the text files to create the CMSVSAM and/or CMSAMS DOSLIBs.	Text files
Create either the CMSVSAM or CMSAMS segments from the associated DOSLIB.	DOSLIBs

The following applies to VSAMGEN functions:

Notes:

1. If this is the initial installation of VSAM, you should choose function 1, 2, or 3.

2. If the text files have already been created from the VSAM distribution tape and are currently on the accessed A-disk, you should choose function 4, 5, or 6.

3. If the DOSLIBs currently reside on an accessed disk, you should choose function 7, 8, or 9. In this case it is not necessary to have the text files available.

Based on the function chosen, VSAMGEN verifies that the necessary input is available. If not, you receive appropriate error messages that identify the specific problem, and processing is terminated.

Invoke the VSAMGEN EXEC procedure by issuing:

vsamgen

VSAMGEN verifies the existence of an accessed, read/write A-disk. If the A-disk is not found, the procedure is terminated and an error message is issued.

The list of functions you can use for installing VSAM and/or Access Method Services is displayed:

```
SELECT ONE OF THE FOLLOWING FUNCTIONS BY ENTERING THE NUMBER:

 1. INSTALL AMS           (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 2. INSTALL VSAM          (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 3. INSTALL VSAM AND AMS  (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 4. BUILD  AMS            (BUILD DOSLIB, CREATE SEGMENT)
 5. BUILD  VSAM           (BUILD DOSLIB, CREATE SEGMENT)
 6. BUILD  VSAM AND AMS   (BUILD DOSLIB, CREATE SEGMENT)
 7. RESTART AMS           (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
 8. RESTART VSAM          (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
 9. RESTART VSAM AND AMS  (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
10. QUIT                  (EXIT VSAMGEN EXECUTION)

       ENTER RESPONSE...
```

While VSAMGEN is processing, you receive error, information, and response type messages. The error and information messages are self-explanatory. Error messages labeled 2101I are information messages from the linkage editor and should be ignored. If you are following the sample DMKSNT shown in Figure 13 on page 111 , respond as follows to the prompting messages.

```
DMSVGN363R ENTER LOCATION WHERE CMSVSAM WILL
           BE LOADED AND SAVED
```

210000

For CMSVSAM, the address must correspond to where DMKSNT expects to load CMSVSAM. **210000** using the sample DMKSNT as shown in Figure 13 on page 111 .

```
DMSVGN366R ENTER NAME OF SYSTEM TO BE SAVED
```

cmsvsam

SAMGEN saves the segment.

```
SGN365I SYSTEM CMSVSAM SAVED
```

Indicates that CMSVSAM has been successfully saved.

```
DMSVGN363R ENTER LOCATION WHERE CMSAMS WILL
           BE LOADED AND SAVED:
```

280000

For CMSAMS the address must correspond to where DMKSNT expects to load CMSAMS. **280000** using the sample DMKSNT as shown in Figure 13 on page 111 .

```
DMSVGN366R ENTER NAME OF SYSTEM TO BE SAVED:
```

cmsams

SAMGEN saves the segment.

```
SGN365I SYSTEM CMSAMS SAVED
```

Indicates that CMSAMS has been successfully saved.

If you wish to save DOSLIB and AMS, you must rename and copy them now.

Continue with Step 4.

Step 4. Load and save CMSDOS

To load and save the CMS/DOS segment called CMSDOS, use the same EXEC procedure (DOSGEN) that you used to load and save the CMS/DOS segment called INSTVSAM in Step 1 on page 112.

If you plan to load the 64K CMS/DOS segment called CMSDOS at 3136K (310000), as in the example shown in Figure 13 on page 111, your virtual machine size must be at least 3328K. Access the CMS system disk as any mode other than S.

Issue the commands:

def stor 4m	
ipl 190	
acc (noprof	
acc 190 b/a dms* text	
dosgen 310000 cmsdos	Saves the CMS/DOS segment.

DOSGEN checks that the address contains valid characters, is greater than X'20000', and less than 16M. If an error is detected, the message

```
DMSGEN095E INVALID ADDRESS 'address'
```

is issued and the command is terminated.

DOSGEN then checks for a read/write A-disk on which to write the CMS loader work file; if none is accessed, it issues an error message and terminates.

```
DMSGEN006E NO READ/WRITE A-DISK ACCESSED
```

Next, DOSGEN loads all the text files needed for VSE simulation. The text files are loaded starting at the address specified on the DOSGEN command. If there are any unresolved external references, DOSGEN terminates execution with the message:

```
DMSGEN111E DOSGEN FAILED DUE TO LOAD ERRORS
```

DOSGEN then assigns a storage key of X'D' to the segment and saves it. If an error is detected, one of the following messages is issued and DOSGEN terminates execution:

```
DMSGEN412S DOSGEN FAILED DUE TO SETKEY ERRORS
DMSGEN141S DOSGEN FAILED DUE TO SAVESYS ERRORS
```

Otherwise, the segment is successfully saved and the load map is written to the A-disk as LOAD MAP A5.

```
SYSTEM SAVED
DMSGEN715I DOSGEN COMPLETE
```

The system name entry in the SYSNAMES table defaults to LOAD.

If you wish to save the load map, access 190 as B and make a copy of the load map now. For example: copy load map a5 cmsdos segmap b2.

At this point you have completed the procedures for loading and saving the discontiguous saved segments that are defined in the DMKSNT module supplied with the Starter System. If you wish to load and save other discontiguous saved segments, you must have created other DMKSNT entries for them, and you now must repeat these procedures for those entries.

Step 5. Generate and Save the CMSL Nucleus Saved Segment

Assuming you are still logged on using the MAINT userid, you can increase your virtual machine storage size, up to the maximum size defined for it in the VM/SP directory, by entering the DEFINE STORAGE command. Then, reload CMS. Issue:

def stor 16m

ipl 190

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

Completes the IPL.

generate cmsl nucleus

The system response will be similar to the following:

```
VMFLOAD BEING EXECUTED
LOADLIST= CMSLOADL AND CNTRL= DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD

THE LOADMAP FOR THE CMS NUCLEUS SHOULD BE SAVED ON MDISK 191,
WHEN THIS STEP COMPLETES, RE-IPL THE CMS SYSTEM (I 190)
AND ENTER: GENERATE MAP
SINCE THE CMS LARGE OPTION HAS BEEN SELECTED
REPLY "NO" TO THE QUESTION "ALSO IPL CYL/BLK 0?"

DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

The response to this prompt is the address of the device where the CMSL nucleus is to be written.

190

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =
```

97

The response to this prompt is device dependent and only applies if the release 3.1 CMS sample minidisk sizes were used. See the table below for the correct CMSL value for your device type.

Device	CMSL
FB-512	44544 blk
3330	097 cyl
3340	229 cyl
3350	047 cyl
3375	072 cyl
3380	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?
```

no

```
DMSINI611R VERSION IDENTIFICATION =
```

You are permitted to enter up to 32 descriptive characters.

ENTER

```
DMSINI612R INSTALLATION HEADING =
```

You are permitted to enter up to 64 descriptive characters.

ENTER

```
VM/SPn xxxxxxxx mm/dd/yy
```

ENTER

```
R;
```

generate map

```
LOAD MAP HAS BEEN PLACED IN READER

SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cmsl

```
'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSLNUC MAP" ON MDISK "191"
R;
```

Note: In the procedure that follows, be sure to enter the SAVESYS CMSL command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMSL system, enter:

ipl 190 97 clear

97 is the cylinder number you used in response to message number DMSINI609R. The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

savesys cmsl

If the CMSL system is named something other than CMSL, see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP RELEASE n  mm/dd/yy hh:mm
```

Your CMSL system is now saved. You can issue IPL CMSL instead of IPL 190, when you wish to run CMSL.

For additional information that may be helpful, see "Loading a CMS Nucleus" on page 176.

Chapter 6. Installation of Optional Feature Program Products

This chapter provides procedures for all new VM/SP users for installing optional feature program products.

The chapter contains:

1. A procedure for formatting and allocating volumes that are not formatted by the PREP EXEC.

2. An overview of program product installation.

3. Information for using the PPPREP EXEC to format the program product-related minidisks.

4. Information for using the INSTFPP EXEC to install the feature program products from the Feature tape(s) and verify that they are installed correctly.

Notes:

1. The optional feature program product execs (PPPREP and INSTFPP) assume that certain disks exist with certain passwords. You should not delete or alter passwords of minidisks related to the optional feature program products you will be installing.

   After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

2. Appendix I, "Optional Feature Program Products" on page 287 contains additional information about the optional feature program products.

3. The sample file listings are in:

   Appendix J, "3310 Based System Sample Files" on page 303
   Appendix K, "3330 Based System Sample Files" on page 333
   Appendix L, "3340 Based System Sample Files" on page 363
   Appendix M, "3350 Based System Sample Files" on page 395
   Appendix N, "3370 Based System Sample Files" on page 425
   Appendix O, "3375 Based System Sample Files" on page 455
   Appendix P, "3380 Based System Sample Files" on page 487

G

Formatting and Allocating Volumes (not done by the PREP EXEC)

The PREP EXEC (using the CPFMT parameter) does not format and allocate DASD volumes such as VMPK02, VMPK03, PROFPK, SQLPK, and SQLPK2. This can be accomplished by loading the "IPL FMT" file to your card reader and then IPLing the card reader to start the FORMAT/ALLOCATE program. This should be done from the MAINT userid. Use the following procedures for each of the volumes to be formatted. (User input is shown as all lowercase characters, system responses are shown in all uppercase characters.)

```
vary on cuu        (cuu = real address of the DASD volume)

CUU VARIED ONLINE
R; T=0.01/0.01 11:39:46

attach cuu *       (cuu = real address of the DASD volume)

DASD CUU ATTACH TO MAINT CUU
R;

pur rdr all

XX FILES PURGED
R;

spool pun *

punch ipl fmt s (noh

PUN FILE XXXX TO  MAINT COPY 001   NOHOLD
R;

ipl 00c

VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:

format

FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):

cuu                (cuu = real address of the DASD volume)

ENTER DEVICE TYPE:

dt                 (dt = device type of the DASD to be formatted.
                    Valid entries are: 3330, 3330-11, 3340-70,
                    3350, 3375, 3380, or FB-512.
                    Use FB-512 for a 3310 or a 3370 system.)

ENTER START CYLINDER (XXX) OR "LABEL":
```

Note: When using FBA devices (3310 or 3370), the system will display ENTER START PAGE (XXX) OR "LABEL."

```
(press ENTER)

ENTER END CYLINDER (XXX):
```

Note: When using FBA devices (3310 or 3370), the system will display ENTER END PAGE (XXX).

```
(press ENTER)

ENTER DEVICE LABEL:

llllll             (enter the label name, ie: VMPK02)

FORMAT STARTED
FORMAT DONE
000 NO. PAGE RECORDS WITH READ-CHECK ERRORS
```

If the previous message indicates other than 000 read-check errors, discontinue the installation process and contact your Systems Engineer or hardware service personnel. (You may have to initialize the volume with the Device Support Facility. Refer to the *Device Support Facility User's Guide and Reference.*)

```
ENTER FORMAT OR ALLOCATE:

 allocate

ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):

 cuu                  (cuu = real address of the DASD volume)

ENTER DEVICE TYPE:

 dt                   (dt = device type of the DASD to be allocated.
                       Valid entries are: 3330, 3330-11, 3340-70,
                       3350, 3375, 3380, or FB-512.
                       Use FB-512 for a 3310 or a 3370 system.)

ENTER DEVICE LABEL:

 llllll               (llllll = label name of DASD volume)

ENTER ALLOCATION DATA FOR VOLUME llllll
TYPE CYL CYL
```

Note: The message above will be "TYPE PAGE PAGE" for FB-512 DASD type.

The following table shows the start and end page/cylinder for each DASD type:

DASD TYPE	3310	3330	3330-11	3340-70	3350	3370	3375	3380
START page/cyl.	00002	000	000	000	000	00002	000	000
END page/cyl.	15751	403	807	695	554	69749	958	884

You must enter the following two lines:

```
 perm  sss  eee       (sss = start cylinder or page for DASD type)
                      (eee = end   cylinder or page for DASD type)
 end

ALLOCATION RESULTS
PERM        XXXXX   XXXXX
DEVICE CUU VOLUME LLLLLL   ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Do not respond to this message. Instead press the PA1 key to enter the CP environment and do the following:

```
 detach cuu           (ccu = real address of the DASD volume)

DASD CUU DETACHED  MAINT CUU

 attach cuu to system as llllll
                      (cuu = real address of DASD volume)
                      (llllll = label name of DASD volume)

DASD CUU ATTACH TO SYSTEM llllll
```

Program Product Installation Overview

Two execs are provided to assist with installation of program products from the Feature tape(s). The execs are the PPPREP EXEC and the INSTFPP EXEC. The PPPREP EXEC is used to format the program product-related minidisks defined in the sample directory. The INSTFPP EXEC is used to install the desired program products.

Appendix I, "Optional Feature Program Products" on page 287 contains a list of the program products that are included on the Feature tape(s). Other product installation related information is also included in the appendix. Refer to the appendix now and as needed during the installation of the program products.

Using the PPPREP Exec for Formatting the Program Product-Related Minidisks

Program product-related minidisks must be formatted before program products can be installed. The PPPREP EXEC is provided to assist in the formatting of the minidisks.

The user who will be formatting the minidisks must be logged on as MAINT. All users should be logged off the system while the minidisks are being formatted because MAINT cannot link to a minidisk in write mode if another user is linked to the minidisk in R/W mode.

Be aware that the PPPREP EXEC will format several minidisks on the system. Any data already on those disks will be destroyed. The minidisks that will be formatted are listed below. The system also prints the list for you during execution of the PPPREP EXEC.

Follow the procedures below to format the minidisks.

1. Logon as MAINT if you are not already logged on with the MAINT userid.

   If you have just completed formatting and allocating any volumes (not done by the PREP EXEC), log off the MAINT userid and log on again. This is to ensure that the minidisk links in the MAINT directory entry will be executed correctly.

2. Force all users, including the operator, except MAINT off the system. This must be done because the PPPREP EXEC links to user minidisks in R/W mode. If another user is linked to a minidisk in R/W mode, MAINT will not be able to make the link in write mode.

   Enter the following commands to notify all users that they should log off the system so that service work can proceed.

   ```
   wng all all users will be forced off the system, log off now
   ```

   The above command will tell any users on the system to log off before you will force them off.

   ```
   query names
   ```

   The above command will display the names of all the users on the system.

   ```
   force xxxxxxxx
   ```

where:

xxxxxxxx is the name of user on system

You must execute the FORCE command for each user on the system, including the operator.

3. Have a copy of the directory handy. If a link fails, you will be asked for the write password of the specific minidisk.

4. Enter the following command to start the formatting process:

```
ppprep
```

Warning: The PPPREP EXEC will format several minidisks on your system. Any data already on those disks will be destroyed. Only use PPPREP prior to the initial installation of optional feature program products.

THE MINIDISKS WHICH WILL BE FORMATTED ARE:

```
ADMIN     191            IPCS     191        OPERATOR 191
AUTOLOG1  191            ISMAINT  191        OPERATNS 191/193
CMSBATCH  195            ISPVM    191/192    OP1      191
CMSUSER   191            MAINT    19E        PVM      191
CPRM      191/192/291    MAINT    300        RSCS     191
DATAMOVR  191            MAINT    310        SYSDUMP1 191
DIRMAINT  191/193/195    MAINT    319        VMAP     191
DISKACNT  191            MAINT    31A        VMUTIL   191
EREP      191            MAINT    325
IIPS      191/193        MAINT    3A0
```

DO YOU WISH TO RUN PPPREP AT THIS TIME? (PLEASE ANSWER YES OR NO)

yes

NOTE: THIS EXEC WILL TAKE SOME TIME TO EXECUTE.

```
FORMATTING MAINT 19E MINIDISK
FORMATTING MAINT 300 MINIDISK
FORMATTING MAINT 310 MINIDISK
FORMATTING MAINT 319 MINIDISK
FORMATTING MAINT 31A MINIDISK
FORMATTING MAINT 325 MINIDISK
FORMATTING MAINT 3A0 MINIDISK
FORMATTING DIRMAINT 191 MINIDISK
FORMATTING OPERATNS 193 MINIDISK
FORMATTING IIPS 191 MINIDISK
FORMATTING IIPS 193 MINIDISK
FORMATTING VMAP 191 MINIDISK
FORMATTING ISMAINT 191 MINIDISK
FORMATTING PVM 191 MINIDISK
FORMATTING ISPVM 192 MINIDISK
FORMATTING CPRM 191 MINIDISK
FORMATTING CPRM 291 MINIDISK
FORMATTING ADMIN 191 MINIDISK
FORMATTING AUTOLOG1 191 MINIDISK
FORMATTING CMSBATCH 195 MINIDISK
FORMATTING CMSUSER 191 MINIDISK
FORMATTING CPRM 192 MINIDISK
FORMATTING DATAMOVR 191 MINIDISK
FORMATTING DIRMAINT 193 MINIDISK
FORMATTING DIRMAINT 195 MINIDISK
FORMATTING DISKACNT 191 MINIDISK
FORMATTING EREP 191 MINIDISK
FORMATTING IPCS 191 MINIDISK
FORMATTING ISPVM 191 MINIDISK
FORMATTING OPERATOR 191 MINIDISK
FORMATTING OPERATNS 191 MINIDISK
FORMATTING OP1 191 MINIDISK
FORMATTING SYSDUMP1 191 MINIDISK
FORMATTING VMUTIL 191 MINIDISK
COPYING SAMPLE PROFILE EXEC TO CMSUSER 191 MINIDISK
FORMATTING RSCS 191 MINIDISK
CON FILE xxxx TO MAINT COPY 001 NOHOLD
```

Note: An error message will be displayed if any of the above minidisks did not format correctly. Investigate why the minidisk did not format correctly and correct the problem. You will then have to format the minidisk manually because if you invoke the PPPREP EXEC again, it will format all the minidisks again. (If you chose to delete the definition for a minidisk which PPPREP tries to format, you will receive a message; you may ignore it.)

Using the INSTFPP Exec for Installing the Feature Program Products

Note: Before using the INSTFPP EXEC to install any optional feature program products, be sure to read "Considerations for Installing Optional Feature Program Products" on page 133.

The INSTFPP EXEC manages the installation of feature program products by doing the following:

- Loads from tape and executes the product installation EXEC.
- Updates the PROD LEVEL file on the MAINT 319 minidisk.
- Prints the product MEMO file and copies it to the MAINT 319 minidisk.

To use the INSTFPP EXEC, you must:

- Log on as MAINT
- Define your storage size as 12M
- IPL CMSL (not CMS).

The INSTFPP EXEC has a help facility that can be invoked by entering HELP INSTFPP.

Have a copy of your directory available. Many of the program product installation execs perform links to the user minidisks in write mode. If the attempt to make the link fails, you may be asked for the write password of the minidisk.

If your printer handles only uppercase characters, you must use the FOLD option of the LOADBUF command. In addition, many of the product memos contain special characters that your printer may not handle. If your printer will not print the special characters, you will have to look at the product memo files online. The product memo files are on the MAINT 319 minidisk.

A console file is created during the installation process to record the terminal activity. This file will be spooled to the MAINT reader at the completion of the INSTFPP EXEC.

The feature program product tape is assumed to be at virtual address 181. To check, enter Q V 181. If the tape is not accessed as 181, access it as 181.

> **Note:** If INSTFPP is invoked with no parameters from a 3270 device (20 line minimum), a set of user assistance panels is displayed.
>
> There are HELP screens available to explain the INSTFPP panels. These screens are invoked when the user presses PF1 from any panel.

INSTFPP command syntax is as follows:

```
INSTFPP    [NOPROMPT] [MEMO] [ALL | prodid( prodid...)]
```

NOPROMPT
eliminates the prompt that requires the user to specify whether or not to install the specified program product.

MEMO
allows printing of the product MEMO files from the tape **without** installing the program products. If MEMO is not used, the program products are installed. The product MEMO file is always printed when a program product is installed. You can find the instructions for manual verification and product installation in the product MEMO files.

ALL
allows the user to process all the program product(s) on the tape. If INSTFPP ALL is entered, all the program products on the tape will be installed, each product MEMO file is printed and the user is prompted before each program product is installed.

prodid(prodid...)
allows the user to specify one or more program product IDs when INSTFPP is invoked. INSTFPP then scans the EXECs on the stacked tape and installs the selected program products.

After all program products are installed, follow the instructions in each product's MEMO file to verify that they were installed correctly. You do not have to do this for products that were automatically verified during installation.

PROD LEVEL File

The INSTFPP EXEC updates a file named "PROD LEVEL" on the MAINT 319 minidisk to indicate the results of each program product installation.

The following is an example of what a PROD LEVEL file may look like:

```
* 5748XXB - DISPLAY MANAGEMENT SYSTEM (DMS/CMS) *
RELEASE 2.0
TIME AND DATE OF ENTRY: 14:22:12 83/10/18
*** PRODUCT INSTALLED AND VERIFIED SUCCESSFULLY

* 5798DMY VM FILE STORAGE FACILITY *
VERSION 1  RELEASE 1  MODIFICATION 0
TIME AND DATE OF ENTRY: 19:16:27 83/10/19
*** PRODUCT INSTALLED; MUST BE MANUALLY VERIFIED
```

Update Messages

Each program product entry in the PROD LEVEL file has an update message associated with it. The possible update messages and their explanations are:

***** PRODUCT INSTALLED AND VERIFIED SUCCESSFULLY**

The program product installed correctly and the product was verified successfully.

***** PRODUCT FILES LOADED; SEE PRODUCT MEMO TO COMPLETE INSTALLATION**

The program product files have been loaded successfully. You must refer to the product memo that was printed out by the INSTFPP EXEC. This memo will tell you how to complete the installation of the product and then verify that it installed correctly. In some cases the product memo may refer to other documentation.

***** PRODUCT INSTALLED; MUST BE MANUALLY VERIFIED**

The program product installed but it was not verified automatically. You must refer to the product memo that was printed out by the INSTFPP EXEC. This memo will tell you how to verify that the program product was installed correctly. In some cases the product memo may refer to other documentation.

***** PRODUCT INSTALLED; VERIFICATION FAILED**

The program product installed but the automatic verification failed. You can try to install the program product again and if it still does not verify correctly, contact your support personnel.

***** PRODUCT INSTALLATION EXEC FAILED, RC = XXX**

The program product installation exec failed. The return code passed back by the program install exec to the INSTFPP EXEC is indicated. Look at the program product installation exec to see what the specific return code means. If you cannot fix the problem, contact your support personnel.

Considerations for Installing Optional Feature Program Products

Some special considerations must be taken into account before several of the program products are installed. These are:

- Many program products will not install properly without CMS Large (CMSL). IPL CMSL before installing optional program products.

- MAINT's 319 (P) and 325 (W) minidisks are NOT able to hold ALL the possible optional program products. You may want to create a separate disk for them. Refer to Figure 26 on page 289 for the space needed for each program product.

- All program products will verify from the CMSUSER userid except for the ones noted here. Refer to the individual product memos for specific instructions on what userid(s) are used for verification. These program products are:

  - CSP/AD (5668-944)
  - CSP/AE (5668-945)
  - DIRMAINT (5748-XE4)
  - EP/VS (5744-AN1)
  - INFO/SYS (5735-OZS)
  - IPF (5748-MS1)
  - PVM (5748-RC1)
  - RSCS (5748-XP1)
  - SMART (5796-PMA)
  - SQL/DS (5748-XXJ)
  - VMAP (5798-CPX)
  - VMBATCH (5796-BCY)

- ADRS-II (5796-PLN) requires VSAPL (5748-AP1) to be installed first.

- ADRS-II (5796-PLN) with 'Full Screen Support' requires GDDM (5748-XXH) to be installed before VSAPL (5748-AP1).

- ADRS-II/BG (5796-PLN) with 'Business Graphics' requires GDDM (5748-XXH) to be installed before VSAPL (5748-AP1).

- APL/DI-II (5796-PNG) requires VSAPL (5748-AP1) and PL/I Transient Library (5734-LM4) to run.

- CSP/AD (5668-944) requires CSP/AE (5668-945) and VSE/VSAM (5746-AM2) to be installed first.

- CSP/AE (5668-945) requires VSE/VSAM (5746-AM2) to be installed first.

- EREP (5749-010) when you verify the installation of this product using the instructions found in the EREP MEMO file, be sure to access the disk containing the V5749010 EXEC as filemode A.

- FPS-II (5798-DCN) requires VSAPL (5748-AP1).

- INFORMATION/SYSTEM (5735-OZS) requires VSE/VSAM (5746-AM2). Installation of the INFORMATION/SYSTEMS-VSE Data Feature requires additional space as specified in the INFORMATION/SYSTEMS Memo to users.

- IPCS/E requires that the CPRM disconnected virtual machine must be logged off before installation of IPCS/E. Users should be informed of this. After the installation and verification of IPCS/E and the creation of IPCSMAPS, the CPRM disconnected virtual machine may be autologged.

  The CP and CMS nucleus maps must be located on the default MAINT minidisks prior to installing IPCS/E. If the nucleus maps were not saved during the base installation, you must save them now. To generate and save the nucleus maps on the default MAINT minidisks, enter the following commands:

  ```
  GENERATE {CP } NUCLEUS
           {CMS}

  GENERATE MAP
  ```

  It may be necessary to allocate a larger OPERATNS 193 minidisk for dump processing depending upon your Virtual Machine storage size. To calculate the needed space, refer to the *IPCS/E General Information*, GC34-2019 manual.

- ISPF/PDF (5764-172) requires ISPF (5668-960) to be installed first.

- PROFS (5664-176) will need the volume PROFPK formatted and allocated before the product can be installed. This product also requires DCF (5748-XX9) to be installed first. You should print and review the PROFS memo file before installing PROFS. Refer to "Formatting and Allocating Volumes (not done by the PREP EXEC)" on page 125 for information about formatting and allocating volumes. Refer to "Using the INSTFPP Exec for Installing the Feature Program Products" on page 130 for information about printing product memos.

- SQL/DS (5748-XXJ) If you are installing SQL/DS and are not using VM/SP sample directory entries, you should refer to the *SQL/Data System Installation - VM/SP*, SH24-5044 manual. That manual describes the virtual machines and minidisks that you must define before loading the SQL/DS distribution tape files to disk.

  If you are using VM/SP sample directory entries, you must format and allocate the SQLPK volume before installing SQL/DS. In addition, if you are using a 3310 or 3340 DASD configuration, you must also format and allocate the SQLPK2 volume before installing SQL/DS. Refer to "Formatting and Allocating Volumes (not done by the PREP EXEC)" on page 125 for information about formatting and allocating volumes.

  When you use the INSTFPP EXEC to load the SQL/DS files from feature tape to disk, the SQL/DS installation exec (I5748XXJ EXEC) is executed. The I5748XXJ EXEC requires write access to the SQL/DS Production Minidisk (SQLDBA 195) and SQL/DS Service Minidisk (SQLDBA 193). To successfully load the SQL/DS files from the feature tape to disk, the MAINT userid cannot already have a 193 or a 195 disk defined. Also, if MAINT has any disks accessed as filemodes V or Q, I5748XXJ EXEC releases them when it begins running and reaccesses them before exiting.

If you are not using the VM/SP supplied sample VM directory entries, you must know the write access passwords assigned for these minidisks. During I5748XXJ EXEC processing, the SQLDBA 193 disk is linked and accessed by the MAINT userid as virtual address 193 and filemode V. This disk is then formatted and the SQL/DS service files are loaded to it. In the same fashion, the SQLDBA 195 disk is linked and accessed by the MAINT userid as virtual address 195 and filemode Q. This disk is then formatted and the SQL/DS production files are loaded to it.

After the SQL/DS distribution tape files are loaded to disk, you must continue with the SQL/DS data base installation process described in the *SQL/Data System Installation - VM/SP*, SH24-5044 manual.

- VM/IFS (5748-XXC) requires VSE/VSAM (5746-AM2), and requires a directory to be added.

- VMAP (5798-CPX) requires PL/I Transient Library (5734-LM5).

- EP/VS (5744-AN1) is supplied on a separate tape. Prior to installing EP/VS you should have completed Step 4 in Appendix D, "Generating and Loading the 3704/3705 Control Program" on page 257. (Because the referenced Step 4 is very time consuming, it is recommended that it be completed before loading the optional feature program products in order to save time while loading the program products.)

  Prior to installing EP/VS you must choose to load the files from the tape to either a permanent minidisk or a temporary minidisk.

  Note: Do not load EP/VS to the CMS system minidisk because doing so could overlay necessary files.

  If you load them to a permanent minidisk, you must allocate space as follows:

DASD TYPE	SPACE REQUIRED
FB-512	30,000 BLOCKS
3330	95 CYL
3340	215 CYL
3350	60 CYL
3375	80 CYL
3380	60 CYL

  If you use a temporary minidisk, the space will be allocated for you.

  Note: The files are loaded from the MAINT userid. The minidisk for these files MUST be accessed as filemode T. If you do not plan on doing the entire generation and loading of the 370X control program at this time, you should load the files to a permanent minidisk. This is because when you log off the MAINT userid the temporary disk is released and the information on it is lost.

  You must now mount the EP/VS tape as 181 and attach it to the MAINT userid then execute the INSTFPP exec to load the files.

After the files are loaded, you must complete the installation. Refer to the MEMO TO USERS for the EP/VS feature program product for more information.

Note: If you have loaded the files onto a temporary minidisk, after loading and generating you may wish to backup the EP/VS disk for future use by using the the CMS TAPE commands.

Note: The above considerations will control the order that you install program products. If you had two tapes and one had ADRS-II (5796-PLN) on it and the other had VSAPL (5748-AP1) on it, you want to make sure that VSAPL is installed first.

Automatic Logon of Disconnected Virtual Machines

You may want to have disconnected virtual machines automatically logged on and running in disconnected mode after a system IPL. To do this, you need to create, or edit, the PROFILE EXEC for the AUTOLOG1 userid. A sample PROFILE EXEC follows:

```
PROFILE EXEC

&TRACE OFF

CP SLEEP 10 SEC

*For ISPF or ISPF/PDF
CP AUTOLOG ISPVM ISPVM

*For PROFS
CP AUTOLOG PROMAIL PROMAIL
CP AUTOLOG PRODBM PRODBM
CP AUTOLOG SFBATCH SFBATCH
CP AUTOLOG SFCAL SFCAL

*For DIRMAINT
CP AUTOLOG DIRMAINT DIRM
CP AUTOLOG DATAMOVR MOVR

*For PVM
CP AUTOLOG PVM PVM

*For RSCS
CP AUTOLOG RSCS RSCS

*For SMART
CP AUTOLOG SMART SMART

*For VM/IPF
CP AUTOLOG CPRM CPRM
CP AUTOLOG DISKACNT ACNT

*For general system use, and VM/IPF, if VM/IPF is installed
CP AUTOLOG CMSBATCH BATCH
CP AUTOLOG VMUTIL VMUTIL
CP LOGOFF

&EXIT
```

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site. You will then need to update the AUTOLOG1 PROFILE with the new passwords.

Part 2. VM/SP Service

About Part 2

Part 2 tells you how to apply corrective service using the system Program Update Tape, Program Temporary Fixes (PTFs), and your own updates to VM/SP. It contains information about the following:

- Introduction
- System Program Update Tape
- A Virtual Machine for Updating VM/SP
- Files for System Updates
- Recommended Procedures for Updating VM/SP
- Updating CP (Building a NEW CP Nucleus)
- Updating CMS and CMSL
- Updating Service Programs
- Updating the Loader Program
- EXEC Procedures and Command Format Summaries.

Note: Some products that have been installed as optional feature program products require product files to be loaded from the feature tapes to DASD before applying service. Refer to the "Memo to User" for each optional feature program product for more information.

Chapter 7. Introduction to VM/SP Service

Updating Your VM/SP System

When you have a servicing task, you want to do it as quickly as possible without excessive delay or unnecessary steps. VM/SP provides you with several procedures and techniques for updating your VM/SP system.

In most cases, servicing of your VM/SP system is done by applying a system Program Update Tape (PUT). The PUT is distributed regularly to each IBM customer. Application is an automatic process carried out by following the detailed instructions included in the PUT Document which comes with every PUT.

There are instances, however, when you need to update an individual module or apply a fix that is not included on the PUT. VM/SP provides you with EXEC procedures and programs to aid you in accomplishing this task. Using a virtual machine, you can do updating and servicing tasks concurrently with other production work.

The framework provided by VM/SP gives you a maximum amount of flexibility in servicing your system. This framework includes:

- A system support plan
- A system Program Update Tape
- Naming conventions for update files and control files
- Several EXEC procedures, commands, and programs. (Figure 16 on page 151 lists the programs.)

Procedures are provided, under CMS, to apply updates to modules and create new load (TEXT) files, to be included in a new CP or CMS nucleus. These procedures maintain the integrity of original source files and previously applied modifications. The update structure involves file naming conventions for update and text files, a set of programs to support the processing, and several EXEC procedures to process the files.

Any non-PUT updates must also be applied to a copy of the original source file. At the same time, any previously applied updates must be repeated. The source can then be assembled and the new text file included in either the CP or CMS system.

All these techniques require the use of CMS. You should have a thorough understanding of the CMS file system and disk search order, the EXEC interpreters, and the UPDATE command before you try to use any of the procedures described.

The *VM/SP CMS User's Guide* provides information on CMS. For reference material on CMS commands and EXEC control statements, refer to the *VM/SP CMS Command and Macro Reference.*

System Program Update Tape (PUT)

IBM regularly distributes a system Program Update Tape (PUT) containing service updates for the VM/370 system control program, VM/SP, and other program products for which you are licensed. Service for VM/SP is cumulative and complete, containing all current and prior service. When applying service to VM/SP, you must *not* apply service to the VM/370 Release 6 IPCS and RSCS components as neither of these products is a discrete component of VM/SP.

VM/System Product service is initiated by the VMSERV EXEC. VMSERV is always supplied in the first tape file of the system PUT. The VMSERV EXEC controls the individual service EXECs that apply service to build a new CP or CMS nucleus from the replacement text decks on the system PUT.

The first file of the system PUT also contains the PUT Document, which describes the PUT and details for the service application process. VMSERV prints the PUT Document and the *Memo to Users.*

The second file on the system PUT contains the *Memo to Users* file(s) for each product that contains service on the PUT volume. This file is loaded on your C-disk and printed by VMSERV. The *Memo to Users* provides step-by-step instructions for applying the updates. It also tells you when you must do additional steps before invoking VMSERV to build a new nucleus. For example, if a macro library has been updated and your system definition files (DMKRIO, DMKSYS, and so on) must be reassembled, the user memo tells you to use the VMFASM EXEC to reassemble the source files.

A Virtual Machine for Updating VM/SP

The VM/SP directory distributed with each Starter System contains an entry for a userid MAINT. You may want to use this userid for system updating and servicing. MAINT's virtual machine should have access to all the disks required for system service.

If you foresee the need to apply your own local updates to VM/SP, you need to add an additional MDISK statement to MAINT's directory entry. The procedure in this section assumes a user defined minidisk address of 295 for local service.

The entries in the VM/SP directory, with the exception of the 293, 294, 393, and 394 virtual disks, are included in the 3330, 3340, 3350, 3375, 3380, and FB-512 VM/SP directories supplied with the Starter System. They should be included in your VM/SP directory, as they are used by IBM for support.

The contents of the following virtual disks are:

Disk	Contents
190	Current CMS system disk
191	Work area
194	CP text retention
19D	HELP files
201	EREP txtlib
293	CMS PTFs, updates, and updated text decks (object modules)
294	CP PTFs, updates, and updated text decks (object modules)
295	User defined minidisk for locally applied service.
393	CMS source and macros

394	CP macros and copy files
cuu	CP system residence device, or a replica of it, for test purposes

These virtual disks are shown in Figure 14.

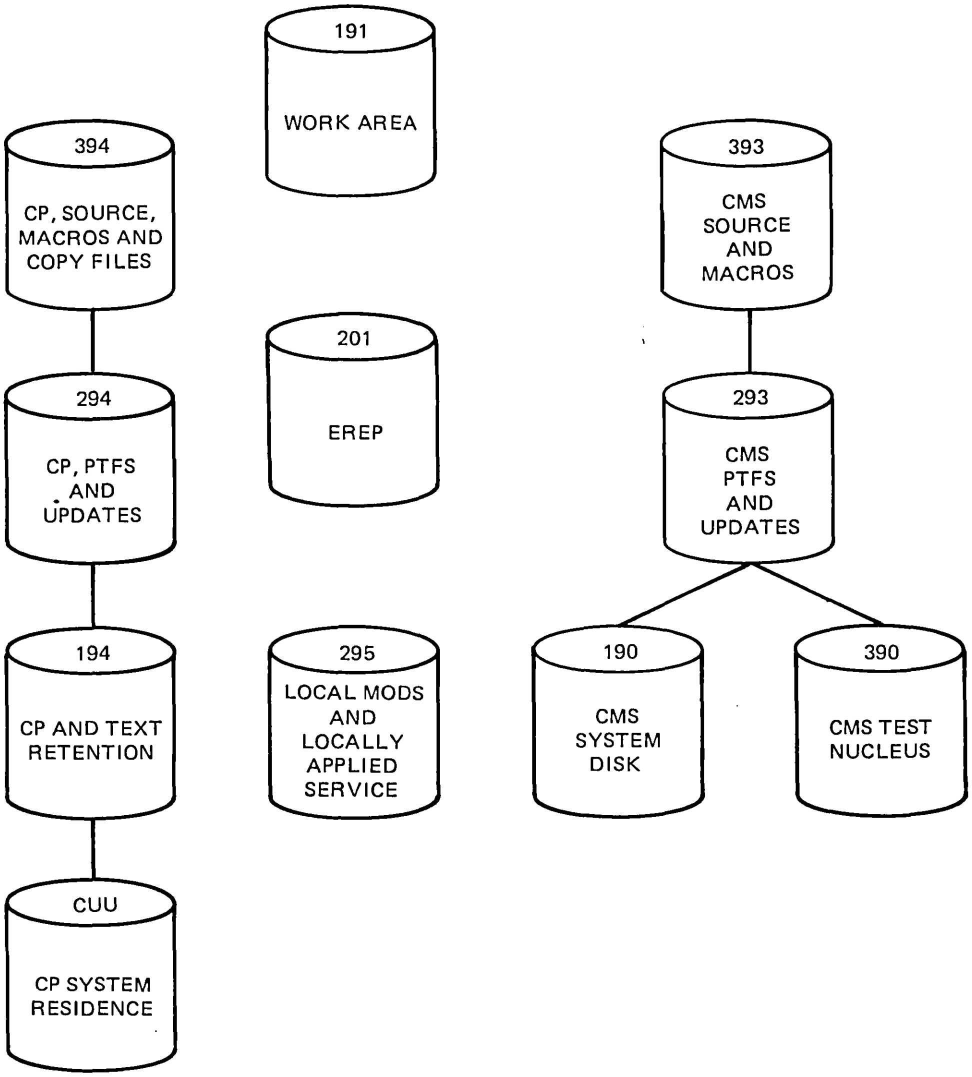


Figure 14. System Support Plan

Suggested VM Configuration for Updating

The following is a sample of a suggested virtual machine configuration for updating. Use the sample for your device type as a guide when applying service.

Example for Updating a 3330 System

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH  A
 SPOOL 00E 1403 A
 MDISK 123 3330 000 404 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 124 3330 000 404 VMSEXT MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3330 000 404 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3330 000 404 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3330 098 100 VMSRES MW ALL     WMAINT  MMAINT
 MDISK 191 3330 021 012 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 194 3330 033 045 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 295 3330 xxx nnn VMSRES MW RMAINT  WMAINT  MMAINT[14]
 MDISK 201 3330 021 045 VMPK01 MW RMAINT  WMAINT  MMAINT
 MDISK 293 3330 335 024 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3330 359 023 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 393 3330 001 130 VMSTGE WR RMAINT  WMAINT
 MDISK 394 3330 234 170 VMSTGE WR RMAINT  WMAINT
 MDISK 19E 3330 195 090 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 319 3330 135 060 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 19D 3330 331 060 VMSRES MW ALL     WMAINT  MMAINT
 MDISK cuu 3330 000 404 yyyyyy MW[15]
```

Figure 15. Suggested VM Configuration for Updating a 3330 System

Files for System Updates

Each of the components of VM/SP has a unique character module identifier, which is used to name the component's modules. These module identifiers are also used to name the files that are used to update the components. The identifiers are:

Component	Module Identifier
CP	DMK
CMS	DMS

The default CMS filetypes are used to identify the source, object code, module files and libraries associated with each component. These filetypes are:

Filetypes	Type of File
ASSEMBLE	Source File
TEXT	Object deck (relocatable)
TXTAP	Object deck with attached processor support (relocatable)
TXTMP	Object deck with multiprocessor support (relocatable)
MODULE	Nonrelocatable object code
MACLIB	Macro or copy library
TXTLIB	Text file library

[14] xxx is the starting address and nnn is the size of the minidisk allocated for your local mods and locally applied service.

[15] cuu and yyyyyy are the address and label of your system residence volume defined in your DMKSYS module.

Control File Identifiers

IBM uses file identifiers as listed below for distributed updates to VM/SP. Use the one CNTRL file that depicts your system configuration.

DMKSP CNTRL: is used for CP source, copy, and macro updates. Its contents are:

```
TEXT MACS DMKSP DMKMAC DMSSP CMSLIB OSMACRO
TEXT AUXSP
```

DMKSPA CNTRL: is used for CP source, copy, and macro updates with support for the attached processor. It should be used for all updates if attached processor support is required. Its contents are:

```
TEXT MACS DMKSPA DMKSP DMKMAC DMSSP CMSLIB OSMACRO
AP   UPDTAP
TEXT AUXSP
```

DMKSPM CNTRL: is used for CP source, copy, and macro updates with support for the multiprocessor system. It should be used for all such updating if multiprocessor support is desired. Its contents are:

```
TEXT MACS DMKSPM DMKSP DMKMAC DMSSP CMSLIB OSMACRO
MP   UPDTMP
AP   UPDTAP
TEXT AUXSP
```

DMSSP CNTRL: is used for CMS source updates. Its contents are:

```
TEXT MACS DMSSP CMSLIB OSMACRO DOSMACRO TSOMAC DMKSP
TEXT AUXSP
```

DMSMSP CNTRL: is used for CMS copy and macro updates. Its contents are:

```
TEXT MACS
TEXT AUXMSP
```

Two of the update procedures, VMFMAC and VMFASM, use the CMS UPDATE command to update macro libraries and source files. Since the updates that are applied are multilevel updates, there are control files (with a filetype of CNTRL) and auxiliary control files, (with filetypes of AUXxxxxx) as well as the actual update files (consisting of UPDATE control statements and new source records). These files may have the following generic filetypes:

Filetype	File Contents
CNTRL	Control file
AUXxxxxx	Auxiliary control file
UPDTxxxx	Local update (listed in a CNTRL file)
anything	Local update (listed in an AUX file)

Note: An AUXSP file may have the release number appended to it as the last two-character component identifier.

Update File Identifiers

All auxiliary control files distributed by IBM have the filetype AUXSP (AUXMSP for CMS MACLIB updates). When an update is issued for a module, an auxiliary control file is also distributed.

All of the PTF and update files distributed by VM/SP are assigned filetypes as follows:

$$\left\{\begin{array}{c}\text{D}\\ \text{F}\end{array}\right\}\begin{array}{l}\text{nnnnnxx}\\ \text{nnnnnxx}\end{array}$$

where

D indicates a System Product module update (or System Product Interpreter EXEC change).

F indicates a System Product CMS macro update.

nnnnn is an APAR or PTF number.

xx is the two-character component identifier (DK or DS).

For example, the code and updates to answer APAR VM12765 against the CP module DMKCFM in the VM/System Product are contained in the file DMKCFM D12765DK. The file DMKCFM AUXSP contains the entry:

```
D12765DK -  COMMENT DESCRIBING FIX
```

When you create files for local corrective service or updates of VM/SP modules, you should create a local control file, that is a copy of the appropriate VM/SP control file with an entry for your AUXLCL auxiliary control file and the filename of your local maclib (if appropriate). For example, the file CPLCL CNTRL may contain:

```
TEXT MACS LCLLIB DMKSP DMKMAC DMSSP CMSLIB OSMACRO
LCL AUXLCL
TEXT AUXSP
```

The IBM-supplied auxiliary files should be the bottom entry in the control file, so that the IBM updates are applied first. (Remember that the UPDATE command, when applying multilevel updates, reads from the bottom of the control file.)

If you plan to update using locally applied service, you should be very familiar with the naming conventions and the files used for updating. Make sure you understand this section before you proceed any further.

Accessing Disks

When you are using the VM/SP procedures to apply updates to system modules, update file identifiers may be duplicated on more than one disk; there may be updates located in several different places. You should always be sure that you have the correct disks accessed, and that you have accessed them with an appropriate search order.

You may find it convenient to create EXEC procedures that do the links and accesses necessary to do a particular update. For example, to update a CP module, from files located on MAINT's 295, 294, 194, and 394, your EXEC procedure might look like the following:

```
ACCESS 295 A    (Local mods)
ACCESS 294 B/A  (CP auxfiles and updates)
ACCESS 194 C/A  (CP text files and maclibs)
ACCESS 394 D/A  (CP assemble, copy, and macro files)
```

This search order ensures that if a control file or auxiliary control file with the same filename exists on both 295 and 294, the one on 295 is used.

VM/SP Integrity

In order to preserve the integrity of VM/SP source files, you should keep updates and PTFs on separate minidisks (not on the same disk as the original source and text files). MAINT's 294 should contain the required IBM PTF updates from the latest PUT. Updates that you make (such as including a PTF obtained from Field Engineering or adding a command to CP), and the resultant text files containing the updates should be kept on MAINT's 295 minidisk.

You also need access to the current CP text files and macro libraries. This is MAINT's 194. This is the disk used by VMSERV when it loads replacement text files from the System PUT.

The CP source files are placed on the 394 minidisk. You should not change these files unless directed to do so by the *Memo to Users*. When you use the CMS UPDATE command and the VMFASM and/or VMFMAC EXEC procedures with the access search order shown in the following examples, changed files are written onto your A-disk[16]. Also, you should not change the IBM-supplied auxiliary files or the PTF files as these are controlled by the PUT procedure. If you want to update a VM/SP component, you should create your own control file on MAINT's 295 minidisk. This file should contain entries for your own updates and for the IBM-supplied updates.

[16] If the A-disk is read only an error message is generated.

Chapter 8. Recommended Procedures for Updating VM/SP

The procedure that you can use to apply local updates are similar for CP and CMS. However, since CMS procedures use different control files, expect the procedures to vary slightly. The examples in the following pages use only CP modules and control files to illustrate the use of:

- The VMFASM EXEC Procedure
- The VMFMAC EXEC Procedure
- The VMFLOAD Program

These, and other procedures and commands are described in "Chapter 10. EXEC Procedures and Command Format Summaries" on page 193.

You should keep in mind that the procedures for updating source files and macro libraries are the same for all VM/SP components, and that the procedure for punching a new CMS nucleus is basically the same as the procedure for punching a CP nucleus.

Local user modifications and locally applied service, as discussed on page 142, should be kept on MAINT's 295 minidisk rather than the 294 minidisk. In this way, you can avoid changing or erasing files that are shipped with the base system or the PUTs.

For specific details and special considerations for loading and testing a new CP or CMS nucleus, or for generating new CMS modules, see:

- "Updating CP"
- "Updating CMS"

The minidisk areas used in the examples in all these discussions use the MAINT virtual machine described under "A Virtual Machine For Updating VM/SP" and illustrated in Figure 14 on page 143. Note that the virtual machine configuration consists of the MAINT entry in the IBM-supplied VM/SP directory, with the addition of an MDISK statement for the system residence volume. Figure 14 on page 143 shows the virtual disks described by the resultant MAINT entry. The following procedures assume that you have also set up a MDISK entry for local mods at address 295. This virtual machine configuration should provide you with all the areas you need to update and test VM/SP.

Deciding Which Procedure to Use

Text level maintenance is available with the system PUT distributed by IBM. When you use this type of maintenance, you do not have to worry about which procedures to use. All updates to a module are applied to the original source file; the source is assembled, and a new TEXT file is sent to you via the PUT. The *Memo to Users* always tells you what to do for the current PUT. Existing TEXT files for your VM/SP system are replaced, on a one-for-one basis, by new files contained on the system PUT. Therefore, it is recommended that you backup your system before applying a new PUT, or create an additional minidisk on which to unload the PUT.

The second type of maintenance involves more work on your part. If you have updates that you want to apply to IBM modules (for example, if you have written an accounting routine you want to include in the DMKACO module), use the following procedures:

1. If an update is being made to a macro library, use the VMFMAC EXEC to update the library.

2. Use the VMFASM EXEC procedure to reassemble the source module using update files. These may be IBM PTFs or updates of your own. If you are reassembling a module because of a MACLIB change, no update files are necessary. A "dummy" update is a convenient way to document those modules assembled for macro updates and achieve correct text file names from the VMFASM EXEC. If you wish to replace a single maclib member, it is recommended that you use the CMS MACLIB command to change the macro library.

3. Use the VMFLOAD program to punch a new CP or CMS nucleus, incorporating existing TEXT files and new ones created by the VMFASM EXEC.

4. Depending on whether you are creating a new CP or CMS nucleus, you may next have to perform additional steps, like writing the new nucleus onto disk, and so on.

Figure 16 below shows the program EXECs available.

Program	Comments
ASMGEND	Updates the VM/370 system assembler.
CMSGEND	Creates a new CMS command module from updated TEXT files.
DOSGEN	Rebuilds the CMSDOS and INSTVSAM discontiguous saved segments from updated TEXT files.
EXECUPDT	Creates an executable version of a System Product Interpreter program after it has been updated using the UPDATE mode of the System Product Editor.
SAMGEN	Rebuilds the CMSBAM discontiguous saved segments from updated TEXT files.
VMFASM	Updates a source file using IBM updates and PTFs and user updates, then assembles the updated source file.
VMFLOAD	Creates a new CP or CMS nucleus based on a control file and a load list EXEC file.
VMFMAC	Updates macro libraries using IBM and user updates.
VMSERV	Controls the individual service EXECs on the system Program Update Tape. Text decks supplied with the PUT replace existing text decks.
VRSIZE	Generates DMKSLC text which is used to generate a V=R area in the system.
VSAMGEN	Rebuilds the CMSVSAM and CMSAMS discontiguous saved segments from updated TEXT files.
VSEVSAM	Builds a VSE/VSAM MACLIB containing the supported VSE/VSAM macros.

Figure 16. Programs for Updating VM/SP

The various procedures and steps to be taken are summarized in Figure 17 on page 152 . These procedures are described in detail in "Chapter 10. EXEC Procedures and Command Format Summaries" on page 193. Before you use any of the procedures, you should have established a virtual machine userid for your maintenance tasks. You must also be acquainted with the CMS files that are used for updating and the naming conventions used by IBM. These topics were discussed in the preceding section headed "Files for System Updates" on page 144.

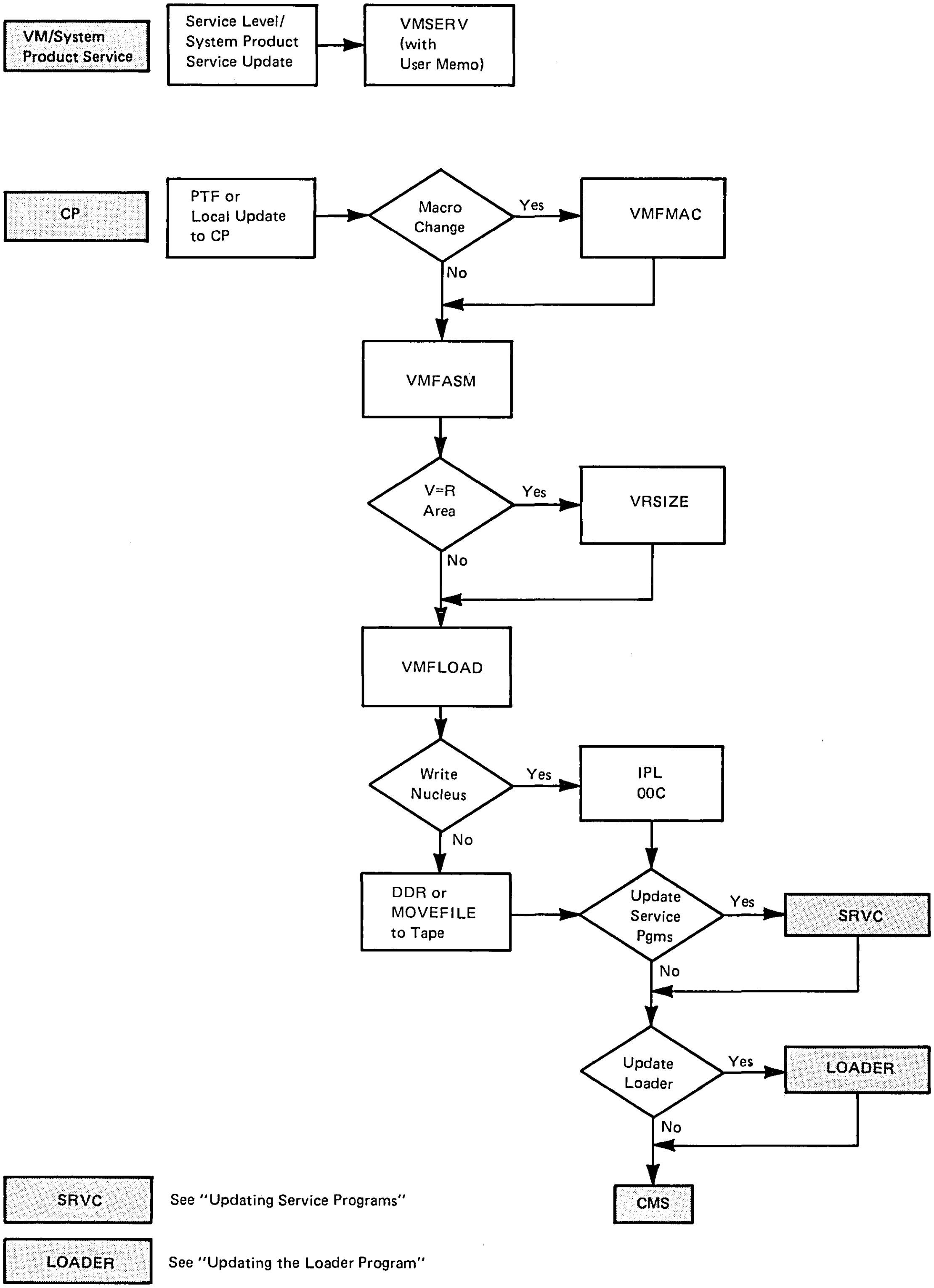


Figure 17 (Part 1 of 2). Deciding Which Updating Procedures To Use

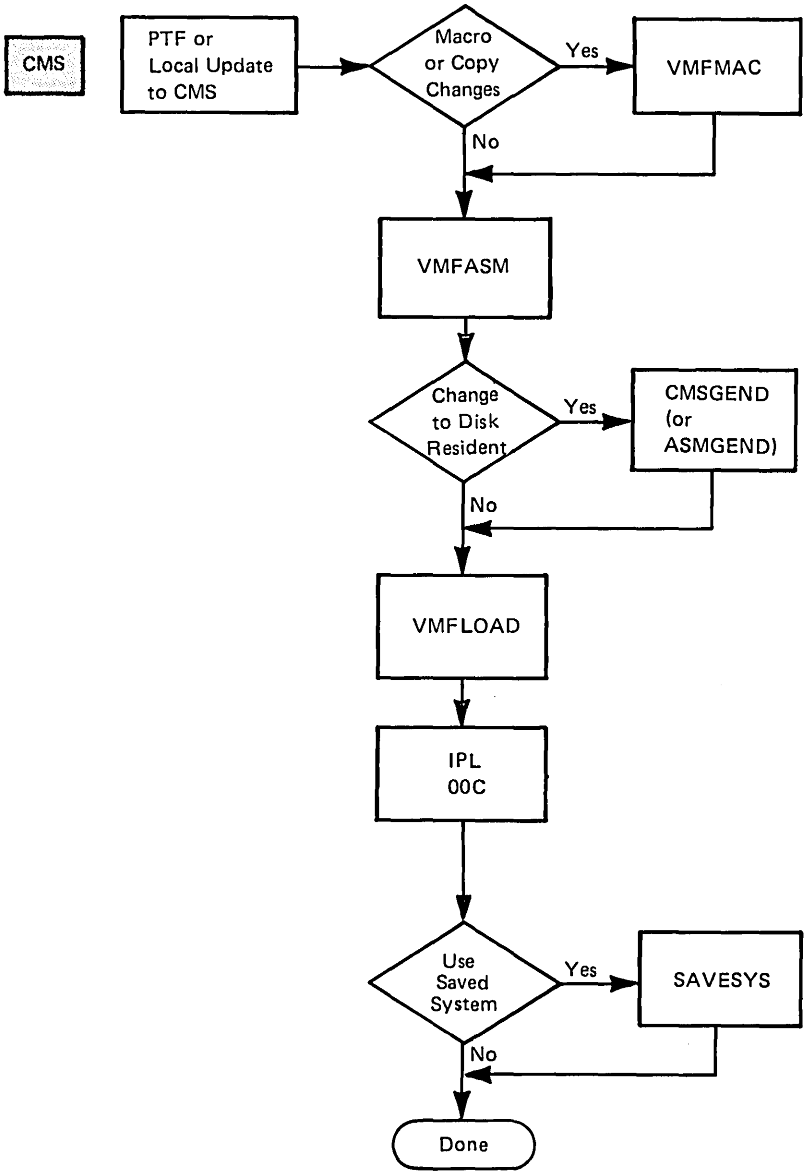


Figure 17 (Part 2 of 2). Deciding Which Updating Procedures To Use

Control File Preparation

Control files are used by the CMS UPDATE command. Both the VMFMAC and VMFASM update procedures invoke UPDATE with the CTL option to modify source files. For VMFMAC and VMFASM, the control file must have a filetype of CNTRL. In addition, the VMFLOAD program also uses a control file: this is usually the same control file used by the VMFASM EXEC.

For an understanding of how the update procedures work, you should have a thorough understanding of the elements in a control file.

Note: The following sample control file is based on changes to DMKSP assuming a uniprocessor system. Remember to change DMKSPA for an attached processor system, or DMKSPM for a multiprocessor system. Control files are described extensively in the *VM/SP CMS User's Guide* and the *VM/SP CMS Command and Macro Reference*. The following discussion summarizes how VMFMAC, VMFASM, and VMFLOAD use the control file.

[1]*THIS IS A SAMPLE CNTRL FILE FOR LOCAL CP UPDATES

TEXT MACS[2] LOCALIB DMKSP DMKMAC DMSSP CMSLIB OSMACRO

UP[3] UPDTFIX1[4]

PTF[5] FIXTEST

LOC1 AUXLCL[6]

TEXT AUXSP[7]

Notes:

1 This is a comment record.

2 VMFASM uses the library list from the MACS record to issue a GLOBAL command before assembling the updated source file. The libraries are searched in the order specified. DMKSPA should precede DMKSP if AP support is required. If MP support is required DMKSPM should precede DMKSP.

3 VMFASM and VMFLOAD use the update level identifier to identify the text deck. VMFASM uses the update level identifier of the most recent update that was found and applied to name the text deck produced by the assembly. VMFLOAD uses update level identifiers to locate text decks when punching a new CP or CMS nucleus.

The update level identifier on the MACS record is used by VMFASM to name an assembled update text deck when no update files are found; it is also used by VMFLOAD when it fails to locate a text file based on update level identifiers associated with update files or auxiliary control files.

4 The characters UPDT identify the filetype of a single update file, UPDTFIX1 in this example. (The characters "UPDT" may be omitted.)

5 The characters PTF in the update level identifier field identify this file as an update file. FIXTEST is the filetype of the update file.

6 The characters AUX identify an auxiliary control file that lists additional updates to be applied; local modifications in this example.

7 AUXSP is the VM/SP auxiliary control file, listing updates distributed by IBM. This file is listed at the bottom of the control file so that these updates are applied first.

A control file can have any number of update identification (UPDTxxxx) records, AUX file identification (AUXxxxxx) records, and comments, but can have only one MACS record.

Example of a CP Update

Suppose that you want to update CP, and then load a new CP nucleus. An assumption has been made that you have allocated and formatted a service minidisk at address 295. The purpose of this minidisk is to store your local, user-written updates and control files and to be able to service your code separately from IBM's. The updates you are going to make consist of the following:

1. You want to add a command to CP. It has already been assembled into the file DMKCMD TXTLCL on MAINT's 295 minidisk. The CP module DMKCFC must be updated to recognize the new command name, so you have updates to apply to DMKCFC.

2. You have a local update to apply to the CP module DMKSCN.

3. You want to change two members of DMKMAC MACLIB, placing the updated members in local maclib LCLMAC MACLIB on MAINT's 295 minidisk. You have updates to apply to ACCTON COPY (for accounting routines) and to RDEVICE MACRO. Since the ACCTON COPY is modified, you have to reassemble DMKACO; changes to the RDEVICE macro require you to reassemble DMKRIO.

The procedures that you would use to do these updates are described next. Remember that the same procedures can be used when you apply updates to any VM/SP components.

Using VMFMAC to Update Macro Libraries

The VMFMAC EXEC procedure is is specifically designed to update macro libraries. You must provide:

- Update files, with UPDATE control statements to change the macro library members. You must also have available any IBM PTFs that have been distributed for the macro library.

- A control file that lists update files or auxiliary control files to be updated.

- An EXEC file listing the names of the members to be included in the macro library.

YOUROWN CNTRL A

TEXT MACS LCLMAC DMKSP DMKMAC
DMSSP CMSLIB OSMACRO
LOC2 UPDTLCL
LOC1 AUXLCL
TEXT AUXSP

LCLMAC EXEC A

&1 &2 RDEVICE MACRO
&1 &2 ACCTON COPY

RDEVICE AUXLCL A

FIXPTF

ACCTON AUXSP C

D12567DK

D12263DK

RDEVICE AUXSP C

D12024DK

Update files

ACCTON D12576DK
ACCTON D12263DK
ACCTON UPDTLCL
RDEVICE FIXPTF
RDEVICE D12024DK

295 (A-disk)

RDEVICE AUXLCL
ACCTON UPDTLCL
LCLMAC EXEC
LCLMAC MACLIB
LCLMAC COPY
YOUROWN CNTRL
RDEVICE FIXPTF

194 (B-disk)

•
•
•
DMKMAC MACLIB
DMKMAC COPY
DMKSP MACLIB
DMKSP COPY

294 (C-disk)

DMKSP CNTRL
ACCTON AUXSP
ACCTON D12576DK
ACCTON D12263DK
RDEVICE AUXSP
RDEVICE D12024DK

394 (D-disk)

•
ACCTON COPY
RDEVICE MACRO

Figure 18. Files for VMFMAC

The files to be used for updating RDEVICE and ACCTON COPY are shown in Figure 18 . In addition to these disks, you should have access to the source COPY and MACRO files on MAINT's 394, and the CMS system disk. The search order should be:

```
295 A     R/W
194 B/A   R/O
294 C/A   R/O
394 D/A   R/O
190 S     R/O
```

If no errors occur during this procedure, VMFMAC erases the existing MACLIB and writes a new one. Ensure the maclibname you are recreating contains the member you are updating. (In this example, you could have accomplished this by copying DMKMAC MACLIB from MAINT's 194 to MAINT's 295 naming it LCLMAC MACLIB).

When you issue the command:

```
      vmfmac lclmac yourown
```

the VMFMAC EXEC procedure uses the LCLMAC EXEC to rebuild LCLMAC MACLIB. VMFMAC calls the UPDATE command to update each of the macro and copy files named in the EXEC.

In this example, ACCTON COPY is updated with YOUROWN CNTRL as follows:

1. The IBM updates named in ACCTON AUXSP (D12263DK and D12576DK) are applied, in that order.
2. Since no ACCTON AUXLCL files exist, the next entry in the control file results in no update.
3. The update file ACCTON UPDTLCL is applied.

For each entry in LCLMAC EXEC, VMFMAC checks to see if there are any updates; if not, then the existing MACRO or COPY file is included in the new MACLIB without changes.

When the entry for RDEVICE is reached, RDEVICE MACRO is updated with YOUROWN CNTRL as follows:

1. The IBM update named in RDEVICE AUXSP (D12024DK) is applied.
2. The update named in RDEVICE AUXLCL (RDEVICE FIXPTF) is applied.
3. Since no RDEVICE UPDTLCL file exists, the last entry in the control file results in no update being applied.

After the entries in the LCLMAC EXEC are processed, VMFMAC erases the existing LCLMAC MACLIB and creates a new LCLMAC MACLIB with the updated members. An additional file, LCLMAC COPY, is produced; this file contains a record of the updates that were applied. LCLMAC COPY is also added to LCLMAC MACLIB, to provide you with a record of changes.

Now, you should create "dummy" updates of the form:

```
   File: DMKRIO FIXPTF
   ./* Indicates reassemble of DMKRIO for RDEVICE FIXPTF
```

If no files with filetype UPDTLCL exist, then the resulting text decks are named TXTLOC1. The filetype indicates which modules need to be reassembled for the local macro "FIXPTF."

Reassemble DMKACO and DMKRIO using the new MACLIB. Use your own file to update them:

```
      vmfasm dmkaco yourown
      vmfasm dmkrio yourown
```

You must be sure that all the current PTFs and auxiliary control files shipped with the base or PUT are available on MAINT's 294, and that all local PTFs and auxiliary control files are available on MAINT's 295. Remember to change DMKSPA for an attached processor system, or DMKSPM for a multiprocessor system.

Using VMFASM to Update Source Files

If you are going to update a VM/SP module, you should always use the VMFASM EXEC procedure, since it allows you to incorporate IBM-supplied updates with your own.

The files in the following example are shown in Figure 19 on page 159. In addition to the 194, 294, and 295 minidisks, you should also have access to the CP assembler language source files on MAINT 394, and the CMS system disk. The search order is:

```
295 A      R/W
294 B/A    R/O
194 C/A    R/O
394 D/A    R/O
190 S      R/O
```

This search order ensures that the DMKCFC AUXLCL file from the 295 is used when the command

```
vmfasm dmkcfc yourown
```

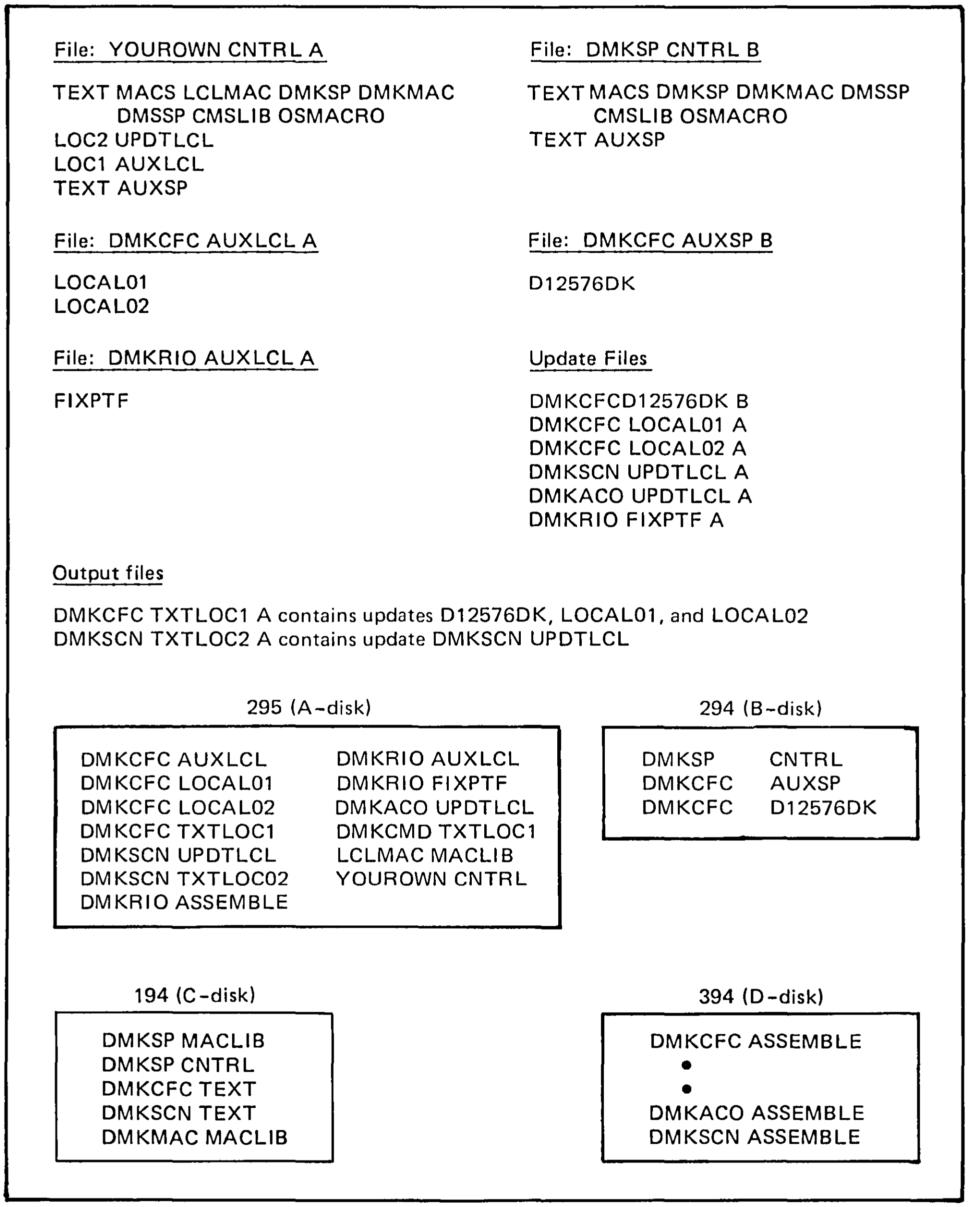
is issued, not a copy on some other disk.

The VMFASM EXEC procedure invokes the UPDATE command with the CTL, STK, PRINT, and OUTMODE A1 options. In this example, UPDATE uses the file YOUROWN CNTRL to determine the order in which to apply the updates. Since the IBM auxiliary control file is the last item in YOUROWN CNTRL, updates named in the file DMKCFC AUXSP are applied first. Then, the updates named in DMKCFC AUXLCL A are applied. Because no file named DMKCFC UPDTLCL exists, no update is applied for that entry in the control file.

When all the updates have been applied, VMFASM calls the assembler to assemble the updated source file, which has a temporary name of $DMKCFC. (If no updates are applied, the temporary name is DMKCFC.) When the assembly is complete, VMFASM uses the update level identifier of the most recent update that was found and applied by the UPDATE command to rename the text file produced by the assembly: in this example, the output file is named DMKCFC TXTLOC1. It is written on the 295 disk, provided 295 is accessed as READ/WRITE.

The updated source file created by the UPDATE command is erased and the original source file remains untouched on the 394 disk.

The UPDATES file produced by a multilevel update is concatenated with the output text deck so that when this object code is loaded, information about its creation is contained in the load map.



File: YOUROWN CNTRL A

```
TEXT MACS LCLMAC DMKSP DMKMAC
       DMSSP CMSLIB OSMACRO
LOC2 UPDTLCL
LOC1 AUXLCL
TEXT AUXSP
```

File: DMKSP CNTRL B

```
TEXT MACS DMKSP DMKMAC DMSSP
       CMSLIB OSMACRO
TEXT AUXSP
```

File: DMKCFC AUXLCL A

```
LOCAL01
LOCAL02
```

File: DMKCFC AUXSP B

```
D12576DK
```

File: DMKRIO AUXLCL A

```
FIXPTF
```

Update Files

```
DMKCFCD12576DK B
DMKCFC LOCAL01 A
DMKCFC LOCAL02 A
DMKSCN UPDTLCL A
DMKACO UPDTLCL A
DMKRIO FIXPTF A
```

Output files

DMKCFC TXTLOC1 A contains updates D12576DK, LOCAL01, and LOCAL02
DMKSCN TXTLOC2 A contains update DMKSCN UPDTLCL

295 (A-disk)

```
DMKCFC AUXLCL       DMKRIO AUXLCL
DMKCFC LOCAL01      DMKRIO FIXPTF
DMKCFC LOCAL02      DMKACO UPDTLCL
DMKCFC TXTLOC1      DMKCMD TXTLOC1
DMKSCN UPDTLCL      LCLMAC MACLIB
DMKSCN TXTLOC02     YOUROWN CNTRL
DMKRIO ASSEMBLE
```

294 (B-disk)

```
DMKSP      CNTRL
DMKCFC     AUXSP
DMKCFC     D12576DK
```

194 (C-disk)

```
DMKSP MACLIB
DMKSP CNTRL
DMKCFC TEXT
DMKSCN TEXT
DMKMAC MACLIB
```

394 (D-disk)

```
DMKCFC ASSEMBLE
  •
  •
DMKACO ASSEMBLE
DMKSCN ASSEMBLE
```

Figure 19. Files for VMFASM

Next, issue the VMFASM command to assemble DMKSCN ASSEMBLE:

```
vmfasm dmkscn yourown
```

The UPDATE command searches for files named DMKSCN AUXSP and DMKSCN AUXLCL. Neither of these files exists; however, DMKSCN UPDTLCL (the local update you created) does exist. This update is applied, the source file is reassembled, and the output file is named DMKSCN TXTLOC2.

The updated source file created by the UPDATE command is erased. The UPDATES file produced by a multilevel update is concatenated with the output text deck, so that when this object code is loaded, information about its creation is contained in the load map.

Notes:

1. VMFASM creates (or replaces, if it already exists) a temporary workfile on the A-disk. For example, if you enter

   ```
   vmfasm dmkjim yourown
   ```

   the work file is named DMKJIM YOUROWN A1. DMKJIM is the assemble-filename and YOUROWN is the control-filename. The work file is erased when VMFASM is finished with it. If you have file with the same name and do not want to lose it, rename the file using the CMS RENAME command.

2. If not enough space is available on your A-disk to contain the assemble workfiles, the following message appears:

   ```
   DMSERD107S DISK 'A(0xxx)' IS FULL
   ```

   and the assembly is terminated.

   To correct the situation, erase any unneeded files from your A-disk or make your A-disk larger. Then, the assemble process can be restarted.

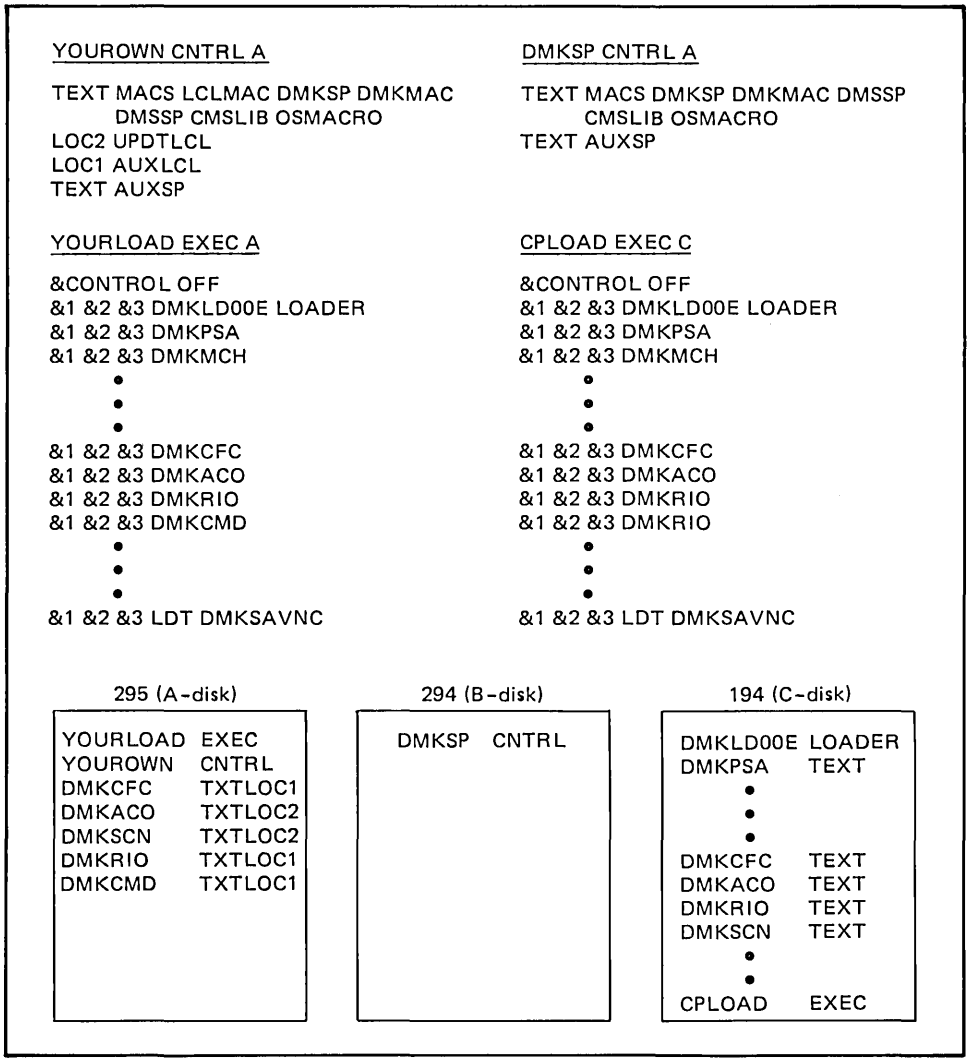
3. To prevent you from inadvertently including a down-level textfile in your nucleus, VMFASM erases any TEXT filetypes on your A-disk for the module you are assembling. A new textfile name other than TEXT, such as TXTLOC1 or TXTLOC2, is erased. For a new textfile name of TEXT, the new file replaces the existing TEXT file on the A-disk.

Using VMFLOAD to Punch a New Nucleus

After you have reassembled all the modules that require updating, you may build a new CP nucleus that contains the updated text decks. In our example, you also want to include your new module, DMKCMD, in the CP nucleus.

To punch a new nucleus, you use the VMFLOAD program, which requires:

- A loadlist file, which must have a filetype of EXEC. It contains the filenames of the object modules in the order in which they are to reside in the nucleus.
- A control file, from which VMFLOAD can determine the filetypes of the latest level text decks, so it can punch them.



```
YOUROWN CNTRL A                               DMKSP CNTRL A

TEXT MACS LCLMAC DMKSP DMKMAC                 TEXT MACS DMKSP DMKMAC DMSSP
      DMSSP CMSLIB OSMACRO                          CMSLIB OSMACRO
LOC2 UPDTLCL                                  TEXT AUXSP
LOC1 AUXLCL
TEXT AUXSP

YOURLOAD EXEC A                               CPLOAD EXEC C

&CONTROL OFF                                  &CONTROL OFF
&1 &2 &3 DMKLD00E LOADER                      &1 &2 &3 DMKLD00E LOADER
&1 &2 &3 DMKPSA                               &1 &2 &3 DMKPSA
&1 &2 &3 DMKMCH                               &1 &2 &3 DMKMCH
      •                                             •
      •                                             •
      •                                             •
&1 &2 &3 DMKCFC                               &1 &2 &3 DMKCFC
&1 &2 &3 DMKACO                               &1 &2 &3 DMKACO
&1 &2 &3 DMKRIO                               &1 &2 &3 DMKRIO
&1 &2 &3 DMKCMD                               &1 &2 &3 DMKRIO
      •                                             •
      •                                             •
      •                                             •
&1 &2 &3 LDT DMKSAVNC                         &1 &2 &3 LDT DMKSAVNC
```

```
      295 (A-disk)              294 (B-disk)              194 (C-disk)

YOURLOAD  EXEC            DMKSP   CNTRL             DMKLD00E  LOADER
YOUROWN   CNTRL                                     DMKPSA    TEXT
DMKCFC    TXTLOC1                                      •
DMKACO    TXTLOC2                                      •
DMKSCN    TXTLOC2                                      •
DMKRIO    TXTLOC1                                   DMKCFC    TEXT
DMKCMD    TXTLOC1                                   DMKACO    TEXT
                                                    DMKRIO    TEXT
                                                    DMKSCN    TEXT
                                                       •
                                                       •
                                                    CPLOAD    EXEC
```

Figure 20. Files for VMFLOAD

The files to be used for creating a new CP nucleus are shown in Figure 20 on page 161. This nucleus incorporates the updates described in the preceding pages. The search order is:

```
295  A    R/W
294  B/A  R/O
194  C/A  R/O
190  S    R/O
```

Since the VMFLOAD program uses your virtual card reader and virtual punch, you must be sure there are no files in either of these devices before you begin. You can issue the commands:

```
close punch
purge punch all
close reader
purge reader all
```

and you must be sure to spool your virtual punch to your own card reader:

```
spool punch *
```

When you issue the command

```
vmfload yourload yourown
```

VMFLOAD uses YOURLOAD EXEC to determine which files to punch. In our example, YOURLOAD EXEC is identical to the distributed CPLOAD EXEC file, except that you have added an entry for your module DMKCMD.

VMFLOAD uses the loadlist to establish the filenames of modules to be punched, and it punches them in the order they appear in the loadlist. Thus, DMKLD00E LOADER is punched first. If a filetype is specified in the loadlist, VMFLOAD punches that file.

When a filetype is not specified (as is usually the case), VMFLOAD uses the update level identifier in the control file to determine the filetype. Since control files are structured so that the most recent update is named at the top of the file, VMFLOAD begins reading at the top of the file.

Since the next entry in the loadlist, DMKPSA, does not provide a filetype, VMFLOAD looks at the control file. In our example, since the update level identifier for the first update record is LOC2, VMFLOAD searches for the file DMKPSA TXTLOC2. Since this file does not exist, VMFLOAD looks at the next lowest identifier: LOC1. It searches for DMKPSA TXTLOC1. Since this file does not exist, it reads the next lowest identifier, TEXT. DMKPSA TEXT exists on the 194 disk, so it is punched. Then VMFLOAD returns to the loadlist EXEC and repeats the procedure for the next entry.

You can see that when VMFLOAD reaches the entry for DMKCFC in the loadlist, it locates the file DMKCFC TXTLOC1, the DMKCFC module that contains your updates.

The loading process continues in this way until the end of the loadlist EXEC file. When all the modules have been punched, you receive the messages:

```
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD
```

These messages indicate that a copy of the new CP nucleus is in your card reader. This CP nucleus contains all the text decks on the 194 disk, except that the files:

```
DMKCFC TXTLOC1
DMKSCN TXTLOC2
DMKACO TXTLOC2
DMKRIO TXTLOC1
DMKCMD TXTLOC1
```

have been punched instead of their text counterparts.

Once the new nucleus has been punched into your card reader, you can load it and test it. Considerations for loading and testing each of the VM/SP components are discussed separately in the following pages.

Chapter 9. Updating An Installed VM/SP System

Building a New CP Nucleus

If you are going to use the MAINT userid to load and test a new CP nucleus, you should be sure that MAINT's virtual machine has:

- A minimum 1M of virtual storage. The loader requires 512K to execute. In general, MAINT's virtual machine should have as much virtual storage as the real machine storage size.

- The ECMODE option specified in the VM/SP directory (or has used the CP SET ECMODE ON command). ECMODE is required for testing the CP system in your virtual machine.

- Write access to the CP system residence volume (SYSRES), or a minidisk that is a replica of SYSRES.

  Notes:

  1. The minidisk must be defined in your virtual machine at the same address as the real address of the SYSRES.

  2. The minidisk must have been formatted with the CP Format/Allocate program, so it resembles the CP system residence.

  3. Unused cylinders beyond the extent of the minidisk must be allocated as permanent space (PERM).

To build a CP nucleus, spool the punch continuous to yourself and invoke VMFLOAD to punch to your reader a file containing the CP loader followed by all the textfiles for the nucleus. Then, IPL the loader file. IPL gives control to the loader. The loader reads in the rest of the deck, link-edits the textfiles, and prints the load map. It then passes control to DMKSAVNC, which writes the nucleus out onto the SYSRES volume described in DMKSYS.

If you followed the steps in the preceding chapter under "Using VMFLOAD To Punch a New Nucleus," the stand-alone loader (DMKLD00E LOADER) is in your card reader. It is followed by all the text decks necessary to build a CP nucleus. There are several ways to handle this reader file.

- Perhaps the most common delivery method is to move the reader file to a tape. Later, IPL the loader from that tape into the real machine and write the nucleus to the real SYSRES.

  To move the reader file to tape, use the following commands:

  ```
  tape rew
  filedef input reader
  filedef output tap1 ( recfm f lrecl 80 block 80
  movefile input output
  ```

- You can get write access to the real SYSRES and IPL the reader file in your virtual machine to write the nucleus out onto your real SYSRES. This is risky because you are writing over the existing nucleus on your SYSRES. If there are errors in your new nucleus, the next time the real system is shutdown and reIPLed you may encounter disastrous and unpredictable results.

- You can set up a virtual SYSRES on a minidisk, giving it the same address and volid as the real SYSRES and CP allocating it. Then, IPL the reader file in your virtual machine to cause the loader to write the new nucleus onto the virtual SYSRES. After you have tested the new nucleus in your virtual machine and are happy with it, you can use DDR COPY NUC command to copy it from your virtual SYSRES to the real one.

  The DDR COPY NUC command might also be used to move the nucleus from the virtual SYSRES to tape. After which you shut the real system down at some point and install the new nucleus using the RESTORE NUC function of stand-alone DDR.

Example of New CP Nucleus Build

For this example, the third method described above is used to handle the reader file. It is assumed that MAINT's virtual machine has entries for the real system residence volume at address 123 and for a minidisk replica at address 331. You need to detach the real SYSRES and define the minidisk at that address by entering:

detach 123

define 331 as 123

Now you can IPL the CP nucleus, specifying the address of your virtual card reader.

ipl 00c

```
NUCLEUS LOADED ON SYSRES
DMKDSP450W CP ENTERED; DISABLED WAIT PSW
CP
```

Load operation complete.
Normal for the stand-alone loader program.

When you IPL the nucleus, the load map is spooled to your virtual printer. To retain a copy of the load map as a CMS disk file, issue the command:

```
  spool printer to *
```

The load map is routed to your card reader. The CMS READCARD command can be used later to write the load map on disk. (Remember to close the spool file.)

Now define your console address to be the same as defined in the RIOGEN macro in DMKRIO.

```
  def 009 as cuu
CONS CUU DEFINED
```

cuu is the console address in RIOGEN macro in DMKRIO.

Then, IPL the replica system residence device by issuing:

```
  ipl 123
VM/SP RELEASE r SERVICE LEVEL nn; mm/dd/yy hh:mm:ss

NOW hh:mm:ss EST weekday mm/dd/yy
CHANGE TOD CLOCK (YES|NO)

  no
hh:mm:ss START ((COLD|WARM|CKPT|FORCE) (DRAIN|SHUTDOWN))

  shutdown
DMKCPI960W SYSTEM WARM START DATA SAVED

DMKCPI961W SYSTEM SHUTDOWN COMPLETE

DMKDSP450W CP ENTERED; DISABLED WAIT PSW
CP
```

After you check the new CP, you may redefine your console by issuing:

```
  def cuu as 009
CONS 009 DEFINED
```

CMS accepts only 009 and 01F as valid console addresses.

IPL the CMS system and issue the DDR command to create a backup copy of the CP nucleus.

```
   ipl cms
CMS...mm/dd/yy
  ddr
ENTER:
   in 123 3380 sysres
```

The input unit is defined using the address of your replica system residence device.

```
ENTER:
   out 181 3420
```

Your output unit is the tape drive that you have attached to your virtual machine at address 181.

When you enter the DUMP statement with the NUCLEUS operand, DDR creates a copy of the nucleus that was just loaded.

```
ENTER:
   dump nuc
DMKDDR711R VOLID READ IS 'volser'
DO YOU WISH TO CONTINUE? RESPOND YES, NO OR REREAD:
   yes
ENTER NEXT EXTENT OR NULL LINE
ENTER:
   ENTER
DUMPING 'volser'
END OF DUMP
ENTER:
   ENTER
END OF JOB
PRT FILE nnnn FOR userid COPY 001 NOHOLD
R;
```

You have created a backup copy of the CP nucleus. This copy may later be restored using the stand-alone version of the DDR program on the real machine.

Updating CMS

The procedure for updating CMS source files and macro libraries is similar to the one for updating CP. However, CMS uses two different control files: one for source files, the other for macros.

The order of search for CMS updates is:

```
295 A   R/W
293 B/A R/O
190 C/A R/O
393 D/A R/O
```

where the 293 minidisk contains PTFs and control files, the 295 contains local mods and/or locally applied service, and the 393 contains the CMS source files. The 190 minidisk contains the current CMS system, including text decks, command modules, and the CMS nucleus.

Note: If you assemble PROP modules DMSPOL, DMSPOP, DMSPOQ, DMSPOR, and DMSPOS, you have to access your 194 minidisk as E/A to gain access to DMKSP MACLIB.

You might use the following steps when you update CMS:

1. Format the minidisk you are going to use to test the CMS nucleus, if any.

2. Update the new text.

3. Use the VMFLOAD program to punch the updated CMS object modules.

4. Regenerate any disk-resident modules that have been updated.

5. Load the new CMS nucleus.

6. Save the new CMS operating system. CMS should be resaved whenever the CMS nucleus or system S-disk directory is updated. This will insure that the saved CMS system reflects the physical system.

The exact steps that you take depend on your individual situation. For instance, the steps to be taken will be affected if you plan to

- Test the CMS nucleus before you load it onto the system disk.

- Use shared segments.

Disks for Updating CMS

If you want to keep CMS source files on disk, the minimum minidisk size to use is shown by device type in the table below.

Device Type	2314	3330	3340	3350	3375	3380	FB-512
Number of cylinders or blocks	261	150	350	72	106	68	68166

Have the CMS source tape mounted and attached to your virtual machine. Then, issue the following commands to load the source programs onto the CMS disk:

```
vmfplc2 fsf

vmfplc2 load (eof 2)
```

If you want to test the new CMS nucleus in a virtual machine before you update the real CMS system, you should have a disk available for a copy of the nucleus. You should change the MAINT virtual machine directory entry to include a 4-cylinder minidisk at virtual address 390 for testing the CMS nucleus. (Or define a temporary minidisk 390 for testing purposes.)

You can test updated disk-resident CMS modules on your A-disk before moving them to the CMS system disk (190).

Formatting a Disk to Test the CMS Nucleus

Before you can use the minidisk you have available for testing CMS, it must be formatted with the CMS FORMAT command. For example, to format the 390 minidisk, you would issue:

```
format 390 g
```

Now, you must reissue the FORMAT command with the RECOMP option, so that the number of cylinders on the disk is recomputed to reserve space for the CMS nucleus at the end of the disk. To do this, format the disk with one or two cylinders fewer than it really has. Use the following table for the value for your device type.

Device Type	190 Sample	Nuc Req	Recomp Value	CMS	CMSL
FB-512	45568 blk	1024	43520	43520	44544
3330	100 cyl	3	94	94	97
3340	236 cyl	7	222	222	229
3350	49 cyl	2	45	45	47
3375	74 cyl	2	70	70	72
3380	45 cyl	1	43	43	44

For example, the 390 minidisk is a 4-cylinder 3330. Thus, you should enter:

```
format 390 g 2 (recomp
```

The 390 disk is now ready for use as the CMS test nucleus. The disk will not have to be formatted again. You can use it each time you update CMS.

Considerations for Creating a New CMS System Disk

If you want to create a new CMS system disk that contains all the CMS text and MODULE files and the CMS nucleus, do the following:

- If you are going to save this CMS system, be sure that the operands VSYSADR, SYSCYL, and VSYSRES in the NAMESYS macro corresponding to this system are correct.

- After copying all the existing files with filetypes of TEXT and MODULE onto the new disk, regenerate any modules that use auxiliary directories (such as the ASSEMBLE command). Auxiliary directories are described in the *VM/SP System Programmer's Guide.* You can use the CMSGEND EXEC procedure to regenerate the assembler. Some IBM Program Products may also use auxiliary directories.

Punching the CMS Nucleus

When you prepare to build a new CMS nucleus, be sure that you have access to the text decks on the system disk, as well as any updated decks that you may have created. Since the CMS text decks are on the CMS system disk (usually 190), you should access it so that you have these text decks available for the VMFLOAD program. To do this, enter:

```
access 190 fm                                fm is a filemode other than S
```

Be sure that your virtual card punch and reader do not have any files in them and that your virtual punch is spooled to your virtual reader. Enter:

```
close punch

purge punch all

close reader

purge reader all

spool punch *
```

Issue the VMFLOAD command specifying the CMS loadlist EXEC filename and the control filename:

```
vmfload cmsload dmssp
```

In this example, the system-supplied CMSLOAD EXEC and DMSSP CNTRL files are used to punch a new CMS nucleus. You will receive the following messages when a new copy of the CMS nucleus is available in your card reader.

```
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD
```

Before you go on to load the new nucleus, you may want to regenerate any CMS MODULE files that have been updated. This procedure is described next.

To decide how to put a given CMS textfile into production, look for its name in the CMSGEND EXEC and the CMSLOAD EXEC (the CMS nucleus loadlist). If the textfile is listed in the CMSGEND EXEC, then it must be incorporated into a CMS MODULE via CMSGEND. If the name is listed in the CMSLOAD EXEC, then the textfile is part of the nucleus. You install it by rebuilding the nucleus with VMFLOAD.

See Appendix A, "CP/CMS Nucleus/Module/Segment Regeneration Requirements" on page 241 for other segments requiring regeneration. If you do not need to regenerate any modules, see "Loading a CMS Nucleus" on page 176.

Creating CMS Disk-Resident Modules

The CMSGEND EXEC procedure creates CMS disk-resident command modules from CMS text files. CMSGEND is invoked by specifying the filename of the module to be generated. For example, if there is a change to the text file DMSACF, you must generate a new ACCESS MODULE.

Access the CMS service minidisk where the updated DMSACF text file exists as your A-disk. Issue:

```
access 293 a
```

Access the 190 system disk as a read-only extension of your A-disk to get access to the CMSGEND EXEC. Issue:

```
access 190 f/a
```

Issue the CMSGEND EXEC to generate a new ACCESS module on your A-disk:

```
cmsgend access
```

After an existing file of ‘ACCESS MODOLD A1’ is erased, CMSGEND renames any existing file from ‘ACCESS MODULE A2’ to ‘ACCESS MODOLD A1’. CMSGEND then loads the text files that comprise the ACCESS command module and generates a new ACCESS MODULE A2.

When you use CMSGEND, you must have all pertinent text files available. The text files must have a filetype of TEXT; thus, if you have updated an object module using VMFASM, and the most recent object file has a filetype such as TXTLOC1, you must rename it to a filetype of TEXT. (Note that if there is currently a text file on the system disk, you may want to rename it also, so that your updated text file, on some other disk, is loaded.)

The following is a sample of what CMSGEND might display when it executes:

```
*** CURRENT STATUS:
FILE ' ACCESS MODULE A2' DOES NOT EXIST
FILE ' ACCESS MODOLD A1' DOES NOT EXIST

*** LOADING:
 INVALID CARD - *    CMSLIB   MACLIB   A2 RnM190 mm/dd/yy hh:mm
 INVALID CARD - *    DOSMACRO MACLIB   A2 RnM190 mm/dd/yy hh:mm
 INVALID CARD - *    DMSACC   ASSEMBLE A1 RnM303 mm/dd/yy hh:mm

 ACCESS   SD 00E000

 INVALID CARD - *    CMSLIB   MACLIB   A2 EnM196 mm/dd/yy hh:mm

 READFST  SD 00EBC0
 DMSACM   SD 00EF10
 READMFD     00EF10

 INVALID CARD - *    OSMACRO  MACLIB   S2 RnM290 mm/dd/yy hh:mm
 INVALID CARD - *    OSMACRO1 MACLIB   S2 RnM290 mm/dd/yy hh:mm

 DMSALU   SD 00F4A8
 RELUFD      00F4A8
 SORTFST     00F716
 END$RELU    00FF38

*** RESULTS:

'ACCESS MODULE A2' CREATED FROM TEXT DECK ( S ) DMSACC DMSACF
DMSACM DMSALU WITH OPTIONS TRANS SYSTEM NOMAP
```

CMSGEND invokes LOAD, which responds with warning messages about comment cards in your textfile, and builds the MODULE. You now have a new ACCESS MODULE on your A-disk which incorporates the updated DMSACF TEXT from the A-disk. The new ACCESS MODULE should now be tested before you move it to the S-disk where everyone can find it.

Loading CMS Disk-Resident Modules

One fact to bear in mind about changing the S-disk or the Y-disk is that the S- and Y-disk directories are saved in the CMS saved system, so you must resave CMS when you change the S- or Y-disks.

Following are the steps you need to take to put your new ACCESS MODULE (from "Creating CMS Disk-Resident Modules," above) into production, starting with the new textfile and MODULE on your A-disk.

Get write access to the 190 minidisk by issuing:

```
link * 190 190 mr

access 190 f
```

To avoid fragmentation on the S- and Y-disks for performance reasons, it is good practice to remove obsolete files. However, do not erase the old textfile on the 190 minidisk since you might need it later.

Rename the old textfile on the 190 making it mode 5. As a mode 5 file, users are not aware of it. Thus, you can erase it when you're sure you no longer need it. To do this, issue:

```
rename dmsacf text f1 = oldtext f5
```

Since the old textfile was mode 1, copy over the new textfile as mode 1. Then, copy over the new ACCESS MODULE under a temporary name. Issue the following commands:

```
copyfile dmsacf text a = = f1 (olddate

copyfile access module a = newmod f1 (olddate
```

Rename the old ACCESS MODULE F2 to ACCESS OLDMOD F5. By making it mode 5, users are not aware of it. Thus, you can erase it when you're sure you no longer need it. Rename the new one ACCESS MODULE F2 by issuing:

```
rename access module f2 = oldmod f5

rename access newmod f1 = module f2
```

To cause the nucleus SSTAT to be rebuilt, you need to IPL 190. (If you IPL CMS, the SSTAT will not be rebuilt.) To rebuild the CMS saved system and incorporate this new SSTAT, immediately issue the SAVESYS command. (Enter the SAVESYS command when CMS puts up its first read, not later, because that is the state in which you want CMS saved.)

```
  ipl 190 clear
VM/SP RELn  mm/dd/yy hh:mm
  savesys cms
VM/SP REL n  mm/dd/yy hh:mm
DMSACC723I Y (19E) R/O[17]
R;
```

Use caution when moving something to the S- or Y-disks. If it is supposed to be mode 1, be sure you make it mode 1. If it is supposed to be mode 2, be sure you make it mode 2. Users will get strange error messages if you make something mode 1 that is really supposed to be mode 2. Remember: The mode of a new file is determined by the mode of the file it is going to replace.

Note: It is recommended that changes to the S- and Y-disks be made with no other users logged onto your system.

[17] This message is displayed only when the Y-STAT is too large to fit in the CMS nucleus. If desired, the CMS nucleus can be generated to allow space for a large Y-STAT. See Appendix F on page 279 for additional information.

Loading a CMS Nucleus

If you are going to test the CMS nucleus on a minidisk other than 190 (we are using 390 is this discussion), you may want to save the nucleus reader file so that you do not have to repeat the VMFLOAD procedure if the nucleus tests out all right. To do this, issue the command:

```
spool reader hold
```

Routes the nucleus load map to your card reader instead of the virtual printer.

```
spool printer *
```

To load the CMS nucleus, issue:

```
ipl 00c clear
DMSINI606R  SYSTEM DISK ADDRESS = cuu
190
```

Enter the device address of the S-disk. This is usually 190.

On this S-disk CMS expects to find all CMS system information and programs not contained within the CMS nucleus, such as the disk-resident command modules. If the CMS nucleus is written on this disk, then **cuu** is also the IPL device address.

- If you enter an invalid device address, the message:

  ```
  DMSINI079E INVALID DEVICE ADDRESS - REENTER
  ```

  is issued. Message DMSINI606R is reissued so that you can enter a valid device address.

- If you press "ENTER" without entering a device address, X'190' is assumed to be the system disk address.

```
DMSINI615R  Y-DISK ADDRESS = cuu
cuu
```

Enter the device address of the Y-disk.

On this disk CMS expects to find all CMS system information and programs not contained within the CMS nucleus and not on the S-disk. If the CMS nucleus is written on the Y-disk, then **cuu** is also the IPL device address.

- If you enter an invalid device address, the message:

  ```
  DMSINI079E INVALID DEVICE ADDRESS - REENTER
  ```

  is issued. Message DMSINI606R is reissued so that you can enter a valid device address.

- If you press **ENTER** without entering a device address, X'19E' is assumed to be the address of the system disk extension.

  Note: If the Y-disk is not wanted, then it should be detached from the userid doing the nucleus build.

```
DMSINI640R HELP DISK ADDRESS =
```

cuu

Enter the device address of the HELP disk.

```
DMSINI607R  REWRITE THE NUCLEUS? (YES|NO)
```

yes

Enter **yes** for a copy of the CMS nucleus to be written onto the disk indicated in the response to message DMSINI608R. Respond **no** here if you are saving more than one nucleus.

- If you enter neither "yes" nor "no," the message:

  ```
  DMSINI081E INVALID REPLY - ANSWER "YES" OR "NO"
  ```

  is issued. Message DMSINI607R is reissued so that you can enter a valid response.

- If you enter **no**, the CMS nucleus is not written to disk. The remaining messages in generating a new CMS nucleus are skipped and control is passed to the CMS initialization routine.

```
DMSINI608R  IPL DEVICE ADDRESS = cuu
```

390

Enter the address of the device on which the CMS nucleus is to be written. (390 for this procedure.) If the system disk and IPL device are to be the same, you need only press **ENTER**.

- If you enter an invalid device address, the message:

  ```
  DMSINI079E INVALID DEVICE ADDRESS - REENTER
  ```

  is issued. Message DMSINI608R is reissued so that you can enter a valid device address.

- If the IPL device you designated is not currently defined, is not in read/write status, or is an unsupported device type, the message:

  ```
  DMSINI082E IPL DEVICE ERROR - REENTER
  ```

  is issued. Message DMSINI608R is then reissued.

  At this time, you may enter CP mode by pressing the ATTENTION key (or equivalent). Then, determine the status of the device you designated by entering the CP command:

  ```
  query virtual cuu
  ```

  and take the corrective action necessary to define the device for your virtual machine or to access it in read/write status. You may re-enter CMS by issuing the CP command:

  ```
  begin
  ```

  Then, you must re-enter the device address. Once the device address is accepted, message DMSINI609R is issued.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS = {nnn|nnnnnn}
```

2

nnn | nnnnn is the cylinder or block number for the device entered in response to message DMSINI608R. Because the nucleus is written on the last cylinder of MAINT's 390 in this example, you enter **2**.

The one- to three-digit cylinder number (nnn) or the FB-512 block number (nnnnnn) is for the device where the CMS nucleus is to be written. The number must be between 1 and m-1. ("m" equals the number of cylinders or blocks on the disk. The cylinders or blocks are numbered from 0 to m-1). The number must be entered in decimal. This is the cylinder or block you reserved when you formatted the disk with the RECOMP option in "Disks for Updating CMS" on page 170.

- If you do not enter a valid decimal cylinder or block number the message:

  ```
  DMSINI080E INVALID CYL/BLK NUMBER - REENTER
  ```

  is issued. Message DMSINI609R is reissued and you can enter a valid cylinder or block number.

- If the cylinder or block specified is not greater than the number of cylinders or blocks already in use on the device (as indicated in the master file directory or volume label), then message:

  ```
  DMSINI083E NUCLEUS CYL/BLK SPECIFICATION
             UNACCEPTABLE, ERROR 'X'
  ```

  is issued.

- Other error conditions and corrective action:

  1. Nucleus will overlap CMS files. Respond to DMSINI609R with a larger cylinder or FB-512 block number.

  2. Nucleus to be written beyond formatted area. Specify a smaller value.

  3. Starting FB-512 block not on 256 block boundary. Specify a number evenly divisible by 256.

  4. Less than 259 FB-512 blocks are available to write the nucleus. Specify a lower location for the nucleus.

  In any event, use of the FORMAT command with the RECOMP option may be necessary.

```
DMSINI610R ALSO IPL CYL/BLK 0? (YES|NO)
```

yes

Initial IPL text is written on cylinder or block 0 and on the cylinder or block designated in response to message DMSINI609R.

The initial IPL text is always written on the same cylinder or block as the CMS nucleus (the cylinder or block designated in response to message DMSINI609R). The initial IPL text is a bootstrap program that reads the nucleus from the designated cylinder or block. If it is not also written on cylinder or block 0, you must enter the cylinder or block number when later IPL commands are issued for the system being generated. See the IPL command description in the *VM/SP CP Command Reference for General Users.*

- If you answer **no** the initial IPL text is written only on the cylinder or block designated in response to message DMSINI609R.

- If you do not enter **yes** or **no**, the message:

  ```
  DMSINI081E INVALID REPLY - ANSWER "YES" OR "NO"
  ```

  is issued. Message DMSINI610R is reissued so that you can enter a valid response. If your new response is valid, message DMSINI611R is issued.

```
DMSINI611R  VERSION IDENTIFICATION =
```

ENTER

Allows you to take the default identification.

You can enter up to 32 bytes of information, including blanks, to specifically identify the version and level of CMS. This information is displayed each time you IPL the CMS system now being generated. The default identification specified by a null entry is:

```
VM/SP REL n mm/dd/yy hh.mm
```

where n is the version and level of CMS, and mm/dd/yy is the month, day and year and hh.mm is the hour and minute the CMS nucleus was created.

```
DMSINI612R  INSTALLATION HEADING =
```

ENTER

Allows you to take the default heading.

You can enter up to 64 bytes of information, including blanks, to serve as an installation standard heading at the beginning of each output file. The default heading specified by a null entry is:

```
VM/SP CONVERSATIONAL MONITOR SYSTEM
```

The nucleus is then written on the specified disk cylinder and the version identification is displayed, indicating that the CMS system is loaded successfully and is ready to accept CMS commands. You can use this copy of CMS to test updates and changes, including changes to CMS modules that you may have made with the CMSGEND EXEC.

Before you test the CMS system, create a disk file from the CMS nucleus and the nucleus load map. Issue the following:

spool reader nohold

close printer

close reader

```
PRT FILE nnnn TO MAINT COPY 001 NOHOLD
```

Now you can read a copy of the CMS nucleus and a copy of the CMS load map onto the 191 disk by entering:

read cmsnuc nucleus a1

generate map

```
LOAD MAP HAS BEEN PLACED IN READER


SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cms

```
'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSNUC MAP" ON MDISK "191"
R;
```

You now have the two CMS files:

- CMSNUC NUCLEUS (contains the CMS nucleus created above).
- CMSNUC LOADMAP (contains the load map for this nucleus).

After you test the new CMS nucleus on 390, and you are satisfied that it is all right, you can use the disk file to create the new nucleus on the system disk (190).

To regenerate a nucleus which exists as a disk file (CMSNUC NUCLEUS. for example), issue the following commands:

```
spool punch to *

punch cmsnuc nucleus a1 (noheader)

ipl 00c
```

You may then answer the IPL messages previously described. This time, you specify the IPL address as 190 instead of 390, and enter the correct cylinder or FB-512 block for your system disk. Then, you can save the CMS saved system, if you wish.

Note: If a named system has been built from this CMS system disk, it must be resaved because the SSTAT is recreated only when the disk is loaded (for example 190).

Saving CMS and CMSL

If your system has entries in the system name table (DMKSNT) for a segment named CMS and a segment named CMSL, you should save these system names now. Be sure that you have defined virtual machine storage to a value above the location of CMS and the loader tables. For example: at least 3M if the DMKSNT locations have been used.

Generating a CMS Nucleus Saved Segment (CMS)

To initiate the GENERATE procedure, issue the following:

```
ipl 190
```

```
DMSACC723I Y (19E) R/O[18]
R;
```

When the GENERATE procedure is completed, you should IPL the CMS system disk again. Then, immediately issue the CP SAVESYS command as follows:

```
ipl 190
```

When VM READ is displayed in the status area of the screen, enter:

```
savesys cms
```

Note: ENTER before issuing the SAVESYS command will complete the IPL too soon.

In this example CMS is the name of the saved CMS system. If you have specified another name for the saved CMS system, you should specify that name when you issue the SAVESYS command. SAVESYS is a CP privilege class E command; it allows you to write on the CP system residence volume.

Now, the saved portion of CMS may be shared among many users, who can load CMS by referring to its saved name, for example:

```
ipl cms
```

When you IPL a saved CMS system, CMS operates as if an IPL of a specific device had occurred, with the exception that the directory for the system disk is part of the nucleus.

[18] This message is displayed only when the Y-STAT is too large to fit in the CMS nucleus. If desired, the CMS nucleus can be generated to allow space for a large Y-STAT. See Appendix F on page 279 for additional information.

Generating and Saving the CMS Nucleus Large Saved Segment

Assuming you are logged on using the MAINT userid, enter:

```
define storage 16m

ipl 190

generate cmsl nucleus
```

The system responds with the following:

```
VMFLOAD BEING EXECUTED
LOADLIST = CMSLOADL AND CNTRL = DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD

THE LOADMAP FOR THE CMS NUCLEUS SHOULD BE SAVED ON MDISK 191,
WHEN THIS STEP COMPLETES, RE-IPL THE CMS SYSTEM (I 190)
AND ENTER: GENERATE MAP

DMSINII606R SYSTEM DISK ADDRESS =

  190

DMSINI615R Y-DISK ADDRESS =

  19e

DMSINI640R HELP DISK ADDRESS =

  19d

DMSINI607R REWRITE THE NUCLEUS ?

  yes

DMSINI608R IPL DEVICE ADDRESS =

  190
```

The response to this prompt is the address of the device where the CMS nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =

  97
```

The response to this prompt is device dependent. See the table below for the correct CMSL value for your device type.

Device	CMS	CMSL
FB-512	43520 blk	44544 blk
3330	094 cyl	097 cyl
3340	222 cyl	229 cyl
3350	045 cyl	047 cyl
3375	070 cyl	072 cyl
3380	043 cyl	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?

  no

DMSINI611R VERSION IDENTIFICATION =

  ENTER
```

You are permitted to enter up to 32 descriptive characters.

```
DMSINI612R INSTALLATION HEADING =
```

You are permitted to enter up to 64 descriptive characters.

ENTER

```
SYSTEM SAVED
VM/SP REL n  mm/dd/yy hh:mm
```

ENTER

```
R;
```

generate map

```
LOAD MAP HAS BEEN PLACED IN READER

SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cmsl

```
'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSLNUC MAP" ON MDISK "191"
R;
```

Note: In the procedure that follows, be sure to enter the SAVESYS CMSL command before **ENTER**. (A null entry usually causes IPL execution to complete.)

To save the CMSL system, enter:

ipl 190 97 clear

97 is the cylinder number you used in response to message number DMSINI609R. The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

savesys cmsl

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP REL n  mm/dd/yy hh:mm
```

Your CMSL system is now saved. You can issue IPL CMSL instead of IPL 190, when you wish to run CMSL.

Updating CMSBAM and CMSDOS

If the CMSBAM or CMSDOS discontiguous saved segments are to be updated, they must be rebuilt in their entirety. Procedures for rebuilding these segments are contained in "Chapter 5. Loading and Saving Discontiguous Saved Segments" on page 109.

Update Procedure for CMS VSAM and Access Method Services

Applying PTFs to the VSE/VSAM Shared Segments

You are responsible for applying service for VSE/VSAM from the VM System Program Update Tape (PUT). This tailored service tape contains VSE/VSAM PTFs in the form of CMS TEXT files and the service EXEC used for applying the service to the CMSVSAM and CMSAMS segments. The master service EXEC (VMSERV) that resides on the service tape will invoke the VSAM service EXEC to apply the PTFs. The EXEC is self-documenting and will prompt you for all information required to restore the updates from the tape and/or rebuild the segments.

If you wish to rebuild the segments after service is loaded, there must be available either a minidisk containing the VSE/VSAM text files that were previously restored from the VSE/VSAM installation tape, or the VSE/VSAM installation tape attached as virtual 182, and a minidisk large enough to hold all the text files. There must also be enough virtual storage defined to allow the various phases to be fetched into storage from the DOSLIBs. If you have followed the example on preparing the System Name Table file (DMKSNT), 8192K is sufficient.

VSE/VSAM Update Considerations

Applying VSE/VSAM PTFs to either CMSVSAM or the CMSAMS discontiguous saved segments may result in the generated segment exceeding the space defined for it in the system name table (see the NAMESYS macro of the DMKSNT file in the *VM/SP Planning Guide and Reference*). You may want to anticipate this problem by defining in the system name table an additional shared and nonshared segment for each of the discontiguous saved segments (CMSVSAM and CMSAMS). This is one way of providing for additional growth.

Alternatively, on completion of the update procedure, you can check whether the updated segments have exceeded their definitions and correct that situation as follows:

1. Determine the new size of the changed VSAM and/or Access Method Services shared and nonshared segments by subtracting the phase LOCORE address from the HICORE address indicated on the linkage editor map. The phase names are:

   - DMSVVS - VSAM shared
   - DMSVVN - VSAM nonshared
   - DMSVAS - Access Method Services shared
   - DMSVAN - Access Method Services nonshared
   - DMSVAX - Access Method Services nonshared

2. Compare the new sizes of these segments with the sizes of the corresponding shared or non-shared segments as defined in your DMKSNT NAMESYS macro.

3. If the new size exceeds your defined size, recode the NAMESYS macro to include an additional segment. Refer to the phase names listed in Step1 to determine whether the segment is shared or nonshared. To add one segment:

   - Increase the SYSPGCT operand by 16

   - Increase the SYSPGNUM operand by 16

   - Increase the SYSHRSG operand by 1, if the segment is shared

   - Increase the SYSSIZE operand by 64K

   - Change the SYSSTRT operand of this or other segments, if the increase in this segment causes any segment to overlap.

4. Reassemble the DMKSNT file, build a new CP nucleus, and then re-execute the VSAMGEN procedure.

If a PTF contains a new VSAM or Access Method Services module, it is not included in CMSVSAM or CMSAMS during VSAMGEN unless you have the current level of installation files.

Updating Service Programs

Service programs are CP modules that are not a part of the CP nucleus. They may run either stand-alone from a card reader (the real system card reader or your virtual card reader) or in some cases, as a CMS command. The service programs are:

- DASD Dump/Restore (module DMKDDR)
- Directory program (module DMKDIR)
- Format/Allocate program (module DMKFMT)
- Device Support Facility
- NCPDUMP, the 3704/3705 dump program (module DMKRND).

If you apply a PTF to a service program, you may use the GENERATE EXEC to create a new IPLable copy of the service program for loading via IPL or the CMSGEND EXEC to create a new CMS command module, or both.

When using the GENERATE EXEC, the text decks being used to create new IPLable copies must reside on the CMS disk.

For example, the Directory program exists as the CP module DMKDIR. To apply PTFs to the source file, you would use the VMFASM EXEC procedure, as follows:

vmfasm dmkdir dmksp	**dmksp** is the filename of the control file (the filetype is CNTRL).

The Directory program can be used as:

- A stand-alone program that you IPL from the real system card reader
- A stand-alone program that you can IPL from your virtual card reader
- A CMS command, DIRECT.

To create a new stand-alone copy via IPL, use the GENERATE EXEC as follows:

```
generate ipldeck
```

You are prompted to enter the name of the program with the following message:

```
ENTER THOSE DECKS TO BE GENERATED ( DDR | DIR | FMT | ALL )
  dir
```

Then, the GENERATE EXEC prompts you to enter the target disk address (where the deck will reside):

```
ENTER TARGET DISK ADDRESS.
  190
```

Accesses the program on the system disk. You must have write access to it.

```
'IPL DIR A2' CREATED
```

GENERATE EXEC is finished.

To generate the CMS DIRECT command, use the CMSGEND EXEC procedure:

```
cmsgend direct
```

If you want to punch a real card deck, to keep available for stand-alone operations in the machine room, you can punch the file IPL DIR A1 (with the NOHEADER option).

Figure 21 lists the services programs and indicates the programs and procedures you can use for each.

Program (With Update)	**DASD Dump/ Restore**	**Directory**	**Format/ Allocate**	**NCPDUMP**	**Device Support Facilities**
CP Module name (use VMFASM to update)	DMKDDR	DMKDIR	DMKFMT	DMKRND	N/A
CMS Command name (use CMSGEND to generate	DDR	DIRECT	——	NCPDUMP	N/A
CMS disk file (use GENERATE IPLDECK to create)	IPL DDR	IPL DIR	IPL FMT	——	N/A

Figure 21. Updating Service Programs

Updating the Loader Program

The loader (DMKLD00E) is a service program that loads a CP or CMS nucleus, and produces a load map. The loader loads the object modules (text files) supplied with it, resolves CCW addresses, and resolves address constants.

If an overlay error occurs while the loader is executing, define a larger virtual machine and reload the system.

The loader is distributed with the following default I/O addresses:

```
Console=009[19]
Printer=00E
```

If there is no printer at address 00E, the load map is printed at the first printer that causes an interrupt (not-ready to ready sequence). A reply of 'ignore' can be made to the printer intervention required message from the loader. This will allow processing to continue without the use of the printer.

[19] The loader does not support display mode consoles. If an IPL is attempted, wait state code 'FFF' occurs if the printer address is not 00E. To circumvent this occurrence, reconfigure the console to printer-keyboard mode or use the override procedure to correct the printer address.

Overriding the Default I/O Addresses

You can override the default I/O addresses by placing a control card between the last card of the loader and the first card of the text decks. The format of the control card is:

Column	Contents
1	12-2-9 multipunch (X'02')
2-4	DEV
5	Blank
6-13	PRNT=cuu (cuu is the printer address)
14	Blank or comma
15-22	TYPW=cuu (cuu is the console address)
23-72	Blank

The other loader control statements are the same as the loader control statements described with the CMS LOAD command in the *VM/SP CMS Command and Macro Reference.*

The loader is self-relocating; that is, it is initially loaded at address 2000 (hexadecimal); it then relocates itself at the top of storage. (For example, if the size of the loader is 10K, and the real storage size of the processor is 512K, the loader occupies the area of storage between 502K and 512K.) As the loader needs free storage to perform its operations, it extends downward through storage.

The object modules being loaded must not overlay either the loader or any address between 0 and 100 (hexadecimal). The object modules are loaded into storage in a positive direction (that is, upward through storage). Before the loader actually loads an object module, it checks that the module does not overlay the loader's free storage. If an object module would overlay the loader, the loader terminates. You must close the printer to get the load map printed. The last line of the load map indicates the overlay area, if there was one.

If the loader terminates the operation, a wait condition is indicated in the instruction counter. If the instruction counter contains X'999999', indicating an SVC wait state, the interruption code (the third and fourth bytes of the supervisor old PSW) indicate the error condition. For a detailed explanation of the error conditions and interruption codes, see *VM/SP System Messages and Codes.*

The Load Map

The load map (the output of the loader) indicates:

- The size of each object module and the address where it is loaded. For example:

  ```
  DMKMCH AT 00E68   MODULE SIZE IS 000C00
  ```

- The end of the resident nucleus with the message:

  ```
  ***                                   ***
        END OF VM/SP  RESIDENT NUCLEUS
  ***                                   ***
  ```

  The CP modules that precede this message in the load map are not pageable; the CP modules that follow this message are pageable.

- When a Set Page Boundary (SPB) card has been inserted. If an object module cannot fit on the same page as the object module(s) loaded before it, the loader inserts an SPB card to force the modules to be loaded at a page boundary. This procedure ensures that object modules do not cross page boundaries.
- Two external names may be listed as undefined on the load map. If the virtual=real option is not specified, the external name DMKSLC is listed as undefined. If a 3704/3705 control program entry is not defined in the system name table (via the NAMENCP macro), the DMKRNTBL external name is undefined.

Generating a New Loader

The loader service program, in its executable form, has a filetype of LOADER. Whenever you assemble a new copy of DMKLD00E, you must convert the resulting text file to a loader file. If there is a virtual punch at address 00D and a virtual reader at address 00C, the procedure for generating a new loader is:

Step 1. Assemble the New Loader

Update and assemble DMKLD00E. The output from this assembly is DMKLD00E TEXT.

Step 2. Punch a Copy of the Old Loader

To spool the punch continuously and punch a copy of the old loader, issue:

```
spool 00d * cont

punch dmkld00e loader (noh
```

Step 3. Punch a Copy of the New Loader Text File

Punch a copy of the newly assembled loader, then close the punch. When the punch is closed the two files (DMKLD00E LOADER and DMKLD00E TEXT) are sent to your reader. The commands to punch the new loader text file and close the punch file are:

```
punch dmkld00e text (noh

spool 00d * nocont close
```

Step 4. Load the New Loader

IPL your virtual reader to read the old version of the loader (DMKLD00E LOADER) into your virtual machine. The old loader reads the new loader text file into your virtual machine and creates the new loader file.

```
ipl 00c clear
```

```
DMKDSP450W CP ENTERED; DISABLED WAIT PSW
```

IPL is complete. The instruction address in the disabled wait PSW is X‘404040’.

Step 5. Punch a Copy of the New Loader (Executable Form)

Close the punch to punch a copy of the new loader, which was created in Step 4. Also, close the reader and printer by issuing:

```
close 00d

close 00c

close 00e
```

Step 6. Name the New Loader DMKLD00E LOADER

IPL CMS and access a read/write disk as your A-disk.

Note: Save a copy of your original DMKLD00E LOADER file before you replace it below with the updated loader. Then, read the file you punched in Step 5 by entering:

```
ipl cms

acc 191 a

read dmkld00e loader
```

By naming this updated file DMKLD00E LOADER, it replaces the original DMKLD00E LOADER file.

Updating Printer Modules

Modify the following as required to meet your installation's needs. See module prologues for specific information.

- Printer Universal Character Set
  DMKUCS (1403)
  DMKUCB (3211)
  DMKUCC (3203)
  DMKPIB (3262)

- Font Offset Buffer
  DMKPIA (3289)

- Forms Control Macros
  DMKFCB (forms control)

To alter any of the printer modules previously listed, enter the following:

```
acc 194 a

acc 394 c

copy dmkxxx assemble c = = a
```

xxx is the appropriate module suffix as listed above.

If the CP source is not on the 394 minidisk, mount the CP source tape on a tape drive at virtual address 181.

Note: If the tape is 6250 bpi, forward space the tape to file 7.

Load DMKxxx by issuing:

```
acc 194 a

vmfplc2 load dmkxxx assemble (eot
```

To modify the file to include your changes enter the following:

```
xedit dmkxxx assemble
```

Make the appropriate changes and store the file. Then, assemble DMKxxx by entering:

```
vmfasm dmkxxx dmksp
```

The DMKxxx file assembles at this time.

If any assembly errors occur, they need to be corrected before the changes you made can be used.

Chapter 10. EXEC Procedures and Command Format Summaries

The command formats, options, and operands for each of the updating EXEC and command procedures are described next, in alphabetical order.

ASMGEND

Use the ASMGEND EXEC procedure to build the system assembler and to create the associated auxiliary directory. ASMGEND loads the text decks for the assembler in the correct overlay structure and produces a load map.

Notes:

1. The assembler text decks normally reside on the system S-disk in file mode S1. This disk must be accessed in some additional file mode prior to issuing this command in order to locate these files. For example:

   ```
   acc 190 a
   ```

2. Use the ASMGEND EXEC if you have modified the assembler (IFOXnn) source. If you have not modified this source, and wish to create the assemble module, possibly after modifying DMSASM or creating a new CMS system disk, use the CMSGEND EXEC.

The format of the ASMGEND command is:

ASMGEND	

Responses

The ASMGEND EXEC procedure displays the following status and error messages:

```
ENTER TARGET DISK MODE FOR ASSEMBLE MODULES
DEFAULTS TO S-DISK IF NONE ENTERED
```

You enter the mode letter of the disk containing the modules referred to from the auxiliary directory. If you enter a mode letter, ASMGEND uses that mode letter as the "targetmode" operand of the GENDIRT command when it creates the auxiliary directory. If you do not specify a mode letter, S is used.

```
ASMGEND XF GEND COMPLETE
```

This message indicates that the system assembler and its associated auxiliary directory are generated successfully.

```
ASMGEND XF GEND FAILED
```

This message indicates that the system assembler text files were not loaded successfully.

CMSGEND

Use the CMSGEND EXEC procedure to generate a new CMS module or LOADLIB from a text file and place the new CMS module or LOADLIB on the specified disk.

Notes:

1. The assembler text decks normally reside on the system S-disk in file mode S1. This disk must be accessed in some additional file mode prior to issuing this command in order to locate these files. For example:

   ```
   acc 190 a
   ```

2. Use the CMSGEND EXEC if you have not modified the assembler (IFOXnn) source, and wish to create the assemble module, possibly after modifying DMSASM or creating a new CMS system disk.

3. You can also use the CMSGEND EXEC to regenerate the ASSEMBLE command when you move the CMS system disk. When you specify ASSEMBLE, CMSGEND prompts you to enter a disk mode letter so it can refresh the assembler's auxiliary directory. (Use the ASMGEND EXEC procedure if you are updating the assembler.)

4. When using CMSGEND EXEC to regenerate the PROP command, you are only generating the PROPLIB LOADLIB. (The CMSGEND options NOCLEAR, MAP, and NOINV have no effect when generating the PROP command.)

The format of the CMSGEND EXEC command is:

```
CMSGEND        fn [ CTLCMS   ]  [ fm ]
                  [ CTLALL   ]  [ A  ]
                  [ NOCLEAR  ]
                  [ MAP      ]
                  [ NOINV    ]
```

where:

`fn` is the filename of the CMS module or LOADLIB that is to be generated by the CMSGEND EXEC. Only one filename may be specified in the CMSGEND command line.

The filenames that may be specified in the CMSGEND command are any disk-resident CMS commands and service programs.

`CTLCMS` displays each CMS command as it is executed in the CMSGEND EXEC procedure. This is equivalent to the EXEC statement &CONTROL CMS.

`CTALL` displays every executable statement as it is executed in the CMSGEND EXEC procedure. This is equivalent to the EXEC statement &CONTROL ALL.

NOCLEAR specifies that the CLEAR option is not to be issued when CMSGEND invokes the LOAD command.

MAP specifies that the NOMAP option is not to be issued when CMSGEND invokes the GENMOD command.

NOINV issues the NOINV option when CMSGEND invokes the LOAD command; this suppresses the displaying of invalid cards at the terminal. If the text deck was created with the VMFASM EXEC, it may contain update listing information; these records are displayed during the loading process unless you specify NOINV.

fm indicates the access mode of the disk to receive the new module. Filemode *A* is the default.

How CMSGEND Works

CMSGEND keeps a list of the CMS disk-resident modules and LOADLIBs, the filenames of the text files used to create them, and any special attributes required to generate them. For example, the RELEASE command must be generated with the ORIGIN TRANS and the SYSTEM options. It is composed of the text files DMSARE and DMSALU. To generate a new RELEASE module, you issue:

```
cmsgend release
```

you may receive messages such as the following:

```
*** CURRENT STATUS:
FILE ' RELEASE MODULE A2' EXISTS
FILE ' RELEASE MODOLD A1' DOES NOT EXIST

*** LOADING:
 INVALID CARD - X9999DMS - (PTF description)
         .
         .
         .
 DMSARE   SD 00E000

*** RESULTS:
FILE ' RELEASE MODULE A2' RENAMED TO ' RELEASE MODOLD A1'
' RELEASE MODULE A2' CREATED FROM TEXT DECK ( S ) DMSARE
WITH OPTIONS TRANS SYSTEM NOMAP
```

Responses

The CMSGEND EXEC procedure displays status and error messages.

```
*** CURRENT STATUS:

['fn MODULE A2' EXISTS        ]
['fn MODULE A2' DOES NOT EXIST]
['fn MODOLD A1' EXISTS        ]
['FN MODOLD A1' DOES NOT EXIST]
```

This message indicates whether a generated module already exists.

```
*** LINK EDITING: fn TEXT
```

This message indicates that CMSGEND is link editing fn TEXT to create a LOADLIB. The existing LOADLIB is erased and not renamed when generating a new one.

```
*** LOADING:
```

This message indicates that CMSGEND is loading the text decks.

```
*** (UNDEF. NAMES NORMAL FOR EDMAIN)
*** NOW WE HAVE A SECOND PASS FOR EDMAIN MODULE.
```

These messages indicate that the EDIT command requires two passes to resolve undefined names.

```
*** NOW WE HAVE A SECOND PASS FOR DMSTPF MODULE.
*** NOW WE HAVE A THIRD PASS FOR DMSTPG MODULE.
```

These messages indicate that the TAPE command requires three passes to generate the necessary modules.

```
*** RESULTS:
['fn MODOLD A1' WAS ERASED                   ]
['fn MODULE A2' RENAMED TO 'fn MODOLD A1'    ]
┌                                             ┐
│'fn MODULE A2' CREATED FROM TEXT DECK(S) ... │
│ WITH OPTIONS ...                            │
└                                             ┘
```

These messages indicate which existing modules were erased and renamed, which text files were used to create the new module, and the attributes used to create the module.

```
ENTER GENDIRT TARGET DISK MODE LETTER
( NULL LINE DEFAULTS TO 'S' DISK )
```

This message is issued when you specify ASSEMBLE. You should enter the mode letter of the disk that contains the system assembler. This letter is used as the target disk mode address for the GENDIRT command.

```
*** ERROR MESSAGE ISSUED IS NORMAL FOR LINK EDITING
```

If the TEXT deck was created with VMFASM EXEC, it may contain update listing information; these records will cause the linkage editor to generate an error message. The error is normal.

Error Messages

```
ERROR OCCURRED. CMSGEND STOPS.
```

This message indicates that an error occurred and that CMSGEND is terminated.

```
INVALID ARGUMENT fn
```

This message indicates an invalid filename was specified on the command line.

GENERATE

GENERATE is a multipurpose EXEC used to generate VM/SP. You may also use it to perform updating and maintenance of:

- CP and CMS
- VM/SP stand-alone service programs

You may also use it to regenerate your VM/SP system after updating:

- The directory
- The real I/O configuration (DMKRIO)
- The system control file (DMKSYS)
- The system name table (DMKSNT)

Instructions for coding the control statements and macros that define your VM/SP directory, and DMKRIO, DMKSYS, and DMKSNT files are in "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference.*

The GENERATE EXEC uses the VMFASM command to assemble the system definition files (DMKSNT, DMKRIO, DMKSYS) and the VMFLOAD command to generate various VM/SP nuclei (CMS/CMSL/CP/CPHPO). These commands use Loadlist and Control files as parameters to determine the selection of module and text names. For additional information on these files, refer to "Chapter 7. Introduction to VM/SP Service" on page 141, "Chapter 8. Recommended Procedures for Updating VM/SP" on page 149, and "Chapter 9. Updating An Installed VM/SP System" on page 165.

An additional control, minidisk access, is also important. The files to be selected/assembled must be available to the user issuing the VMFLOAD/VMFASM command. This is done by having the desired files accessed by the CMS Virtual machine. The search order starts at the A-disk by CMS convention and other minidisks would be accessed after the A (B,C,etc..). In this manner it is possible to have duplicate named TEXT files accessed and control the selection of the proper one.

The result of the VMFLOAD command is a loadable file in the reader of the issuer. This file, when IPl'd will result in the loading of a new system nucleus (assuming proper access to disks).

The GENERATE EXEC without a GENERATE DEFAULTS file has default values hardcoded for the VMFLOAD/VMFASM. These values are determined by interactive prompt except for the minidisk defaults which are:

CP	- 194 minidisk to filemode A
CMS/CMSL	- 190 minidisk to filemode A

Creation of a GENERATE DEFAULTS file allows the override of these hardcoded defaults.

The GENERATE EXEC procedure assumes the following:

Virtual CP nucleus tape address = 182[20]
Virtual address of CMS build area = 190
Virtual address of CP build area = 194
Virtual card reader = 00C

The format of the GENERATE EXEC command is:

```
GENERATE    { ACCESS [DEFINE]                                             }
            { VMSP [HPO]                                                  }
            { DIRECT [ONLY] [HPO]                                         }
            { DMKRIO [ONLY] [HPO]                                         }
            { DMKSYS [ONLY] [HPO]                                         }
            { DMKSNT [ONLY] [HPO]                                         }
            { IPLDECK                                                     }
            { { CP    } NUCLEUS [NOLOAD] [PASS] [minidisk]                }
            { { CPHPO }                                                   }
            { { CMS   }                                                   }
            { { CMSL  }                                                   }
            { TEMP [HPO]                                                  }
            { MAP                                                         }
            { HELP                                                        }
            { EXIT                                                        }
```

where:

ACCESS [DEFINE]

allows the use of a default parameter list file named GENERATE DEFAULTS to be used to determine a installation minidisk access search order for building of a CMS or CP nucleus. If this command is issued prior to service updates, the a-disk will contain the correct code level when the nucleus is built.

The define option creates the GENERATE DEFAULTS file.

VMSP [HPO]

builds a new VM/SP directory, assembles DMKRIO, DMKSYS, and DMKSNT. It also writes the CP nucleus to tape and optionally loads it.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

This option sets up the minidisk search order based on the GENERATE DEFAULTS file definition or uses the defaults of 194 - a and 191 - c/a. If your files are to be on a unique mdisk, access this as d/a to force text decks to be written to the 194 mdisk. The first file encountered in the search order will be executed. If an error occurs during one of these steps, the routine will exit after the step currently being processed. If no errors occur, the routine will go on to generate a new cp nucleus.

[20] DMKSYS, SYSRES = cuu parameter must be defined as the physical DASD address in order to work on a real machine.

You must have the appropriate files in your minidisk search order:

'fn' DIRECT (Directory program control statements)
DMKRIO ASSEMBLE (Real I/O Configuration Macros)
DMKSYS ASSEMBLE (CP System Control Macros)
DMKSNT ASSEMBLE (System Name Table Macros)

where 'fn' is the filename of your VM/SP directory. (GENERATE prompts you for your directory filename.)

GENERATE invokes the Directory program to build the VM/SP directory. It invokes the VMFASM EXEC procedure to assemble the ASSEMBLE files. VMFASM is invoked with the current IBM-supplied control file to ensure that the proper macro libraries are available when the modules are assembled and to assign the correct filetype. Then, GENERATE invokes the VMFLOAD EXEC procedure to place all the CP object modules on tape in the correct order. If an error is detected during any of these processing steps, GENERATE terminates at the end of that step.

For a CP nucleus without a virtual=real area, GENERATE loads the tape, thus loading the newly generated CP nucleus. For a CP nucleus with a virtual=real area, GENERATE writes the nucleus to tape and exits. You are instructed to shutdown the system. Then you can IPL the tape on a real machine or on a virtual machine that has enough virtual storage. "Specifying a Virtual=Real Machine" in Part 1 of the *VM/SP Planning Guide and Reference* tells you how much virtual storage you need.

```
DIRECT [ONLY] [HPO]
```

builds a new VM/SP directory.

If you do not specify ONLY, GENERATE executes exactly as if you specified GENERATE VMSP.

If you specify ONLY, only the VM/SP directory is built. GENERATE prompts you for the filename of your directory file.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

```
DMKRIO [ONLY] [HPO]
```

assembles the real I/O configuration file (DMKRIO) by invoking VMFASM.

If you do not specify ONLY, the GENERATE EXEC procedure executes just as if you specified GENERATE VMSP. It assembles DMKSYS and DMKSNT and generates a CP nucleus (but does not build a new VM/SP directory). Consequently, you should follow the directions for issuing GENERATE VMSP.

If you specify ONLY, only the DMKRIO module is assembled.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

```
DMKSYS [ONLY] [HPO]
```

assembles the CP system control file (DMKSYS) by invoking VMFASM.

If you do not specify ONLY, the GENERATE EXEC procedure executes just as if you specified GENERATE VMSP. It assembles DMKSNT and generates a CP nucleus (but does not build a new VM/SP directory or assemble a new DMKRIO). Consequently, you should follow the directions for issuing GENERATE VMSP.

If you specify ONLY, only the DMKSYS module is assembled.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

```
DMKSNT [ONLY] [HPO]
```

assembles the system name table (DMKSNT) by invoking VMFASM.

If the ONLY option is not specified, it also generates a CP nucleus. The GENERATE EXEC procedure goes on to invoke the VMFLOAD EXEC procedure to place all the CP object modules on tape. If no error occurs, and if the CP nucleus does not have a virtual=real area, GENERATE then loads the tape, thus loading the CP nucleus (with the new version of DMKSNT).

If you specify ONLY, only the DMKSNT module is assembled.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

```
IPLDECK
```

creates the stand-alone service programs on disk from their associated object modules (text decks). These files must be on the CMS system disk (S-disk). You are prompted to enter the names of the service programs you wish to generate. You can respond ALL, DDR, DIR, or FMT. If you respond ALL, the DASD Dump Restore, Directory, and Format/Allocate stand-alone programs are built on disk.

```
( CP    )NUCLEUS [NOLOAD] [PASS[21]] [minidisk[21]]
) CPHPO (
) CMS   (
( CMSL  )
```

generates the specified nucleus.

If you specify CP NUCLEUS or CPHPO NUCLEUS, the CP nucleus is loaded onto tape or written to the target disk.

Note: Using default samples this tape will not work first level unless the DASD is cuu- 123.

For a CP nucleus without a virtual=real area, GENERATE loads the tape, thus loading the newly generated CP nucleus. For a CP nucleus with a virtual=real area, GENERATE writes the nucleus to tape and exits. You are instructed to shutdown the system. Then, you can IPL the tape on a real machine[22] or on a virtual machine that has enough virtual storage. "Specifying a Virtual=Real Machine" in Part 1 of the *VM/SP Planning Guide and Reference* tells you how much virtual storage you need.

Attached Processor support will be included in the nucleus, if desired.

If you specify CMS NUCLEUS, a card-image deck is created and placed in the virtual card reader.

GENERATE issues a prompting message to see if you want a card-image copy of the CMS nucleus put on disk. If you respond "yes," GENERATE writes a copy of the CMS nucleus on disk, and loads the nucleus. The card-image copy of the CMS nucleus is a file (CMSNUC NUCLEUS) that can later be loaded to create a CMS nucleus.

If you specify CMSL NUCLEUS, the saved CMS named system is stored at a higher location (just below 16M instead of just below 2M).

The option NOLOAD, results in the nucleus's not being IPL'ed. In the case of CMS, the new nucleus will remain in the card virtual card reader if the 'noload' option is in effect. If you specify CP NUCLEUS NOLOAD, the tape is created with the new nucleus but the disk is not loaded.

[21] The parameters 'PASS' and 'minidisk' are used with VM/IPF. This represents a minidisk to be accessed as filemode "a" when creating a CP or CMS nucleus. MDISKs 194 or 190 will be accessed as filemode "g/a." The minidisk must be the fifth parameter. When executing generate with less than four parameters fill parameters "pass" must be specified.

For example: generate cms nucleus pass pass 191

[22] DMKSYS, SYSRES = cuu parameter must be defined as the physical DASD address in order to work on a real machine.

`TEMP [HPO]`

formats and allocates a temporary minidisk as 192 for assembler work space. If you have a 192 disk currently attached to your virtual machine, you may want to redefine it. It then continues the GENERATE VMSP procedure.

If you specify HPO, the GENERATE DEFAULTS file is used to determine the loadlist, control file, and access order.

`MAP`

reads and names the nucleus load map from the reader.

A prompt for a nucleus map prefix is issued. If no specific minidisk is specified, CMS and CMSL maps default to the 191 minidisk. All other nucleus maps default to the 194 minidisk. To change the default, reply with the desired minidisk after the map prefix, for example: CMS 194. This will put CMSNUC MAP on minidisk 194.

`HELP`

prints out the prologue of the GENERATE EXEC and describes all of the available options.

Responses and Error Messages

The GENERATE EXEC procedure issues many descriptive responses, most of which are shown in the system generation procedures in "Part 1. VM/SP Installation" on page 1.

The return codes that you will receive are:

RETURN CODES	**ERROR**
1	Invalid operand for generate exec
2	Invalid operand for ipldeck option
6	Error creating directory
8	Error encountered during assembly
10	Error reading cms nucleus to disk
36	Target disk a is read only
99	Undefined access parameter
100	Target disk address not attached

Create A GENERATE DEFAULTS File

Generation of the GENERATE DEFAULTS file is optional for base VM/SP installation but required for HPO installation.

To create the GENERATE DEFAULTS file you will do the following:

- Issue GENERATE ACCESS DEFINE

This will produce a panel with default values for loadlist, control file, and access order as follows.

```
      CNTRL    LOAD          MINIDISK ACCESS ORDER
NUCLS FILE     LIST     A    B    C    D    E    F    G    H
----- -------- -------- ---------------------------------------
CP    DMKSP    CPLOAD   194  PASS 191  PASS PASS PASS PASS PASS
CPHPO DMKH34   H34CPLOD 196  PASS 191  194  PASS PASS PASS PASS
CMS   DMSSP    CMSLOAD  190  PASS 191  PASS PASS PASS PASS PASS
CMSL  DMSSP    CMSLOADL 190  PASS 191  PASS PASS PASS PASS PASS
```

- Modify panel entries to generate desired nucleus if the defaults are not appropriate.

  A Help file is available with the IBM supplied combinations.

- File (PF5) the panel to create the GENERATE DEFAULTS file.

Using the GENERATE DEFAULTS File as A Service Tool

Use the GENERATE EXEC nucleus build features to isolate service and/or user modified code from the base code. This example creates a new CP nucleus and assumes that service update and text files have been loaded to the MAINT 294 minidisk with the service exec.

Preparation Step

Define the generate defaults access order to be used by doing the following:

- Issue GENERATE ACCESS DEFINE

  This will produce a panel with default values for loadlist, control file, and access order. The following example is for CP files.

```
      CNTRL    LOAD          MINIDISK ACCESS ORDER
NUCLS FILE     LIST     A    B    C    D    E    F    G    H
----- -------- -------- ---------------------------------------
CP    DMKSP    CPLOAD   194  PASS 191  PASS PASS PASS PASS PASS

    DEFAULTS FOR SERVICE ACCESS
_____ DUMMY    DUMMY    PASS PASS PASS PASS PASS PASS PASS PASS
```

- Modify panel entries to create defaults for service access

  The following example shows a use of the access file definition which would allow isolation of user, service and base files.

  Minidisk search address usage:

A-disk (295)	Highest priority minidisk. It contains all user modules.
B-disk (192)	R/W minidisk used for ASSEMBLER workspace (created with the GENERATE TEMP option)
C-disk (191)	Sample files and GENERATE DEFAULTS file.
D-disk (294)	New CP service update and text files.
E-disk (194)	Base text files, remain unchanged.
F-disk (394)	Base source files, used for reassembly w/updates.

These minidisks would be inserted into the generate defaults file as follows to create a fixed search order.

```
CPSV DUMMY    DUMMY    295  192  191  294  194  394  PASS PASS
```

- Modify panel entries to generate desired nucleus.

  This step will put all user and service updated files in a higher priority access than the base files. This allows isolation of fixes/modifications and a record of generate exec usage. Note that minidisks 295, 294 and 191 will be accessed prior to the base 194 minidisk.

  The A-minidisks from the previous step would be inserted as the A-minidisks of the CP definition. All other minidisks would be shifted to a lower priority. The B-disk may be left open if the generate temp option is to be used since that option creates a temporary 192 minidisk and accesses it as B.

```
      CNTRL    LOAD              MINIDISK ACCESS ORDER
NUCLS FILE     LIST      A    B    C    D    E    F    G    H
----- -------- -------- ---------------------------------------
CP    DMKSP    CPLOAD    295  PASS 191  294  194  PASS PASS PASS
```

Note: The 394 minidisk is not accessed because a nucleus build does not require the assemble files to be accessed.

- File panel modifications (PF5).

- You will remain in the GENERATE EXEC.

- Issue ACCESS CPSV

  The newly defined search order will now be in effect and should contain all of the updates plus the source files (394) required to create any new text decks. The GENERATE EXEC will now be exited.

Execution Step

- Issue VMFASM fn ctlfile

  All the newly created text files will be put on the A-DISK (295).

- Issue GENERATE CP NUCLEUS

  The newly defined CP search order will now be in effect for use in building a new CP nucleus with all modification/service included.

Other levels of minidisk isolation can be implemented by following these examples.

DISKMAP

The DISKMAP EXEC summarizes the MDISK statements in a CP directory. The output of the exec shows gaps and overlaps between minidisk assignments. The syntax of the DISKMAP exec is:

```
DISKMAP        filename     [filetype]
                            [DIRECT  ]
```

where:

filename   is the filename of the directory to be mapped.

filetype   is the filetype of the directory to be mapped. The default is DIRECT.

The output from the DISKMAP EXEC is a file sent to your A disk. The filename of the output is the same as that of the target directory. The filetype of the output is DISKMAP. The file contains information on MDISK statements found in the directory. The files are in order by volume in the output file. Gaps between minidisks and overlapping minidisks are flagged.

DISKMAP does NOT replace the EDIT function of the DIRECT command. You should use both to check your directory after changes. (The DIRECT command is described in the *VM/SP Planning Guide and Reference*.)

Notes:

1. Because some DASD types come in several sizes, DISKMAP does not list gaps found after all minidisks. You will need to know the maximum cylinder/block value for your DASD type.

2. DISKMAP creates both the map and a workfile on your A disk. If your directory is very large and your A disk is almost filled, you may need to find some extra disk space in order to run DISKMAP.

3. You may choose to include some overlaps in the directory. DISKMAP flags ALL overlaps; you must understand your layout to determine if a particular overlap is expected or in error. For instance, in the VM/SP sample directory MAINT and SYSDUMP1 have full-pack minidisks defined (12X) for backup purposes.

Example for Using DISKMAP

In order to see how DISKMAP works enter:

DISKMAP VMUSERS

This will produce a directory map of the sample files that you loaded during installation of VM/SP. This map should be similar to the directory map sample for your DASD type as shown in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample Files" on page 333

Appendix L, “3340 Based System Sample Files” on page 363
Appendix M, “3350 Based System Sample Files” on page 395
Appendix N, “3370 Based System Sample Files” on page 425
Appendix O, “3375 Based System Sample Files” on page 455
Appendix P, “3380 Based System Sample Files” on page 487

PRELOAD

The preloader is a utility program that runs under CMS. It collects multiple text files and reformats them into a single text file. The function of the preloader is similar to that of a linkage editor, but the output is in standard text file format and does not include multiple CSECTS.

A program can be developed using separate assembly modules that reference each other. The preloader can then be used to combine the assembled text files into a single loadable text file.

The preloader is invoked by issuing:

`PRELOAD`	`loadlist   [control]`

where:

loadlist specifies the filename of an EXEC on the caller's A-disk or read-only extension containing records that define input to the preloader. Each of these records contains the filename and optionally the filetype of an input text file. The format of each loadlist record defining an input file must be:

```
&1 &2 filename  filetype
```

control optionally specifies the filename of a CNTRL file residing on the caller's A-disk or read-only extension. The format and interpretation of the CNTRL file is the same as that for the VM/SP VMFLOAD utility. It normally contains filetypes in priority sequence to be used for selecting input files if filetypes are not included in the loadlist file.

Note: PRELOAD ignores records that have a PTF update level identifier. It then searches for the next lower level identifier to determine the filetype of the input text file.

Input

The preloader gets input filenames from the loadlist. The filetype for each input file is determined in one of three ways:

1. If the loadlist record for a given input file includes a filetype entry, that filetype is used to locate the record.

2. If the loadlist record does not contain a filetype, and a 'control' parameter was specified on the PRELOAD command line, the filetype constructed is in the format 'TXTxxxxx'. In this case 'xxxxx' is the highest control level identifier in the control file for which a file can be located on the caller's accessed disks.

3. If no filetype is specified on the loadlist entry, and a control file has not been specified on the PRELOAD command line, then the default filetype value is TEXT.

Note: Input files are located by scanning the caller's disks in their access order. All input files must be on the caller's A-disk or read-only extension.

Output

The preloader output consists of two files written to the caller's A-disk:

1. fn TEXT
2. fn MAP

The filename for each of these files is the same as that specified for the input loadlist file. If either of these files already exists on the caller's A-disk, the new copy replaces the old one.

TEXT File

The output TEXT file is a merged and linked composite of the input files. The first CSECT or private code section in the input expands to contain all input files. Its length is the sum of the lengths of the input files, rounded up to doubleword multiples between sections. Input TXT records of non-zero length are relocated and written to the output file.

The output RLD is a translated and relocated collection of all input RLD records. No sorting is done by the preloader. In general, each output ESD, TXT, and RLD entry appears in the same order as the corresponding input entry. ADCON and VCON fields are relocated within their TXT records. ORG statements that cause relocatable constant fields to overlay or be overlaid may cause results that differ from results obtained with a loader that completes TXT data loading prior to relocating ADCONs and VCONS.

MAP File

The output MAP file is a printable record of preloader processing, similar to a load map. The first line of the map contains:

- Output text filename
- Residence volume label and volume device address
- Date and time of file creation

The next section of the map is a listing of the control file (if any) used. The remainder of the map contains, in processing order, a section for each input file. Each of these sections consists of:

- Filename, filetype, filemode of input file
- Residence volume label and virtual device address
- Input file's creation date and time
- Any invalid input records

Messages

PRELOAD issues the following CMS messages:

DMSPRE001E	NO FILENAME SPECIFIED
DMSPRE002E	FILE 'fn ft fm' NOT FOUND
DMSPRE104S	ERROR 'nn' READING FILE 'fn ft fm' FROM DISK
DMSPRE105S	ERROR 'nn' WRITING FILE 'fn[ft] fm ' ON DISK
DMSPRE1095	VIRTUAL STORAGE CAPACITY EXCEEDED
DMSPRE183E	INVALID CONTROL FILE CONTROL CARD
DMSPRE234E	ERROR IN LOAD LIST FILE 'fn ft fm' NO INPUT
DMSPRE235E	ERROR 'n' IN INPUT TEXT FILE 'fn[ft] fm '
DMSPRE236E	UNRESOLVED EXTERNAL REFERENCE(S) ENCOUNTERED
DMSPRE237E	DUPLICATE EXTERNAL SYMBOL(S) ENCOUNTERED
DMSPRE238E	PRELOADER PROCESSING ERROR

These messages are documented in *VM/SP System Messages and Codes*.

PREP

Use the PREP EXEC to assist you in preparing the Starter System to become your running VM/SP system. The Starter System volume (VMSRES) layout allows this installation process to format, load, and create your VM/SP system.

Notes:

1. PREP EXEC must be executed on the C accessed minidisk if not on 191.
2. Much of the manual generation that was required for installation in Release 1, has been eliminated by PREP EXEC. PREP EXEC automates the steps previously needed to generate VM/SP. For further insight into exactly what the PREP EXEC does, print a copy of it and follow the GENERATE procedure in more detail.

The PREP EXEC:

- Prepares additional DASD volumes required for a 2- or 3-volume system.

  Two-DASD system device types are 3350, 3370, 3375, and 3380.

  Three-DASD system device types are 3310, 3330, 3330-11, and 3340-70.

- Loads the "default" sample system assemble files, as well as the CP directory. These files are compatible with your Starter System device type.

  DMKRIO ASSEMBLE
  DMKSNT ASSEMBLE
  DMKSYS ASSEMBLE
  VMUSERS DIRECT

  The loaded Directory and DMKxxx files are ready to use, but you can tailor them before creating a VM/SP system. (Update DMKRIO to specify your system's I/O hardware configuration.)

- Formats the minidisks required to load the VM/SP Product Tape.

  MAINT 194 (CP TEXT deck disk)
  MAINT 19D (CMS HELP file disk)
  MAINT 293 (CMS auxiliary and update files)
  MAINT 294 (CP auxiliary and update files)

- Aids in loading the VM/SP source code to the 393 and 394 minidisks (MAINT).

- Aids in creating a stand-alone utilities tape that can be initialized.

The format of the PREP EXEC command is:

```
PREP        ( CPFMT                                  )
            |          ┌                 ┐           |
            | CMSFMT   │ NOSAMP (ONLY)   │           |
            |          │ SAMP ONLY       │           |
            |          └                 ┘           |
            |        ┌┌        ┐       ┐            |
            | LOAD   ││ DEFINE │  NUC  │            |
            {        ││ NODEF  │       │            }
            |        │└        ┘       │            |
            |        └                 ┘            |
            | SAMPLE                                |
            | LOADSRC                               |
            | UTIL                                  |
            | DOHELP                                |
            | HELP                                  |
            ( HPO                                   )
```

where:

CPFMT formats and allocates volumes, other than VMSRES, used for creating and maintaining your system. You are given a choice of up to three volumes, but due to Starter System limitations, only two can be selected during the initial installation.

CAUTION
This function will destroy the contents of these disks if they currently exist. This operand should be used only if you are using the Starter System.

CMSFMT formats the following Starter System minidisks:

- MAINT 194
- MAINT 293
- MAINT 294
- MAINT 19D

The CMSFMT operand allows for additional parameters to be used:

- The NOSAMP option formats the required minidisks. No sample files are loaded and the directory is not updated.

- The NOSAMP ONLY option formats the required minidisks. No sample files are loaded.

  Note: If your SYSRES volume is a 3370 device, the DIRECT 3370 sample from the product tape will replace the current directory. This is normally a 3310 layout on starter systems. To prevent this replacement, issue the CMSFMT NOSAMP ONLY option.

- The SAMP ONLY option loads samples from user tapes.

CAUTION
This function will destroy the contents of these minidisks if they currently exist. This operand should be used only if you are using the Starter System.

LOAD loads the Product Tape (tape drive 181) onto the following minidisks:

- 194 minidisk (CP files)
- 19D minidisk (HELP files)
- 190 minidisk (CMS files).

If 190 has been loaded, then the ASMGEND EXEC is executed to recreate the assembler module. The 190 MDISK will then be reIPLed automatically.

The LOAD operand allows two additional options for loads on non-Release 2 systems. The DEFINE Option allows an override of the default MDISK load. The NODEF Option allows you to keep the default MDISK load. The NUC parameter provides for building a CMS nucleus at the Release 3 level.

Note: Selecting the NUC parameter requires that either DEFINE or NODEF also be specified.

SAMPLE allows the option of printing the sample files that are loaded by the installation EXEC (PREP LOAD).

LOADSRC loads CP and/or CMS source from tape (6250 or 1600 bpi) to minidisks (MAINT 393 AND 394).

UTIL writes to tape as separate files, the stand-alone programs that can be IPLed. The choices are:

- DSF (Device Support Facilities)
- DIR (CP directory program)
- FMT (CP format/allocate program)
- DDR (DASD dump/restores program).

DOHELP erases current HELP files on the 190 minidisk, or changes HELP files to uppercase representation.

HELP prints out the prologue of PREP EXEC and describes all of the available options.

HPO loads the HPO installation exec from the tape at virtual address 181.

Invokes the HPO installation exec to prepare and load the HPO minidisks and files.

Responses and Error Messages

The PREP EXEC procedure issues many descriptive responses, most of which are shown in the system generation procedures in "Part 1. VM/SP Installation" on page 1.

VMFASM

Use the VMFASM EXEC procedure to update a specified source file according to entries in a control file, and to assemble the updated source file. VMFASM invokes the CMS UPDATE command. The format of the VMFASM command is:

```
VMFASM          fn ctlfile [(options...)]

                Options:

                ┌      ┐ ┌      ┐ ┌      ┐
                │DISK  │ │TERM  │ │LIST  │
                │PRINT │ │NOTERM│ │NOLIST│
                └      ┘ └      ┘ └      ┘

                ┌      ┐ ┌      ┐                  ┌   ┐
                │DECK  │ │RENT  │  [EXP]  [XREF]   │MAX│
                │NODECK│ │NORENT│                  │MIN│
                └      ┘ └      ┘                  │STD│
                                                   └   ┘
```

where:

`fn` is the filename of the source file to be updated. It must have a filetype of ASSEMBLE.

`ctlfile` is the filename of the control file. The control file must have a filetype of CNTRL.

Options: VMFASM only accepts the nondefaulted options. All other assembler options entered are ignored and the defaults are used.

`DISK` places the listing file on a virtual disk.

`PRINT` writes the listing file to the printer.

`TERM` writes the diagnostic information on the SYSTERM data set. The diagnostic information consists of the diagnosed statement followed by the error message issued.

`NOTERM` suppresses the TERM option.

`LIST` produces an assembler listing.

`NOLIST` does not produce an assembler listing.

`DECK` writes an object module on the device specified on the FILEDEF statement for PUNCH.

`NODECK` suppresses the DECK option.

`RENT` checks the program for a possible violation of program reenterability. Code that makes the program nonreenterable is identified by an error message.

`NORENT` suppresses the RENT option.

EXP expands printing of certain macros which check for the SUP parameter issued via the SYSPARM option of the ASSEMBLE command. The default is SUP.

XREF causes the XREF(FULL) option to be invoked when VMFASM invokes the assembler. The default for VMFASM is XREF(SHORT).

MAX causes the BUFSIZE(MAX) option to be invoked when VMFASM invokes the assembler. If both MAX and MIN are specified, then MIN takes precedence.

MIN causes the BUFSIZE(MIN) option to be invoked when VMFASM invokes the assembler. If both MAX and MIN are specified, then MIN takes precedence.

STD causes the BUFSIZE(STD) option to be invoked when VMFASM invokes the assembler. This is the default for VMFASM when neither MAX nor MIN is selected.

How VMFASM Works

The steps taken by the VMFASM EXEC are summarized below.

1. The VMFASM EXEC calls the UPDATE command with the CTL, STK, PRINT, and OUTMODE A1 options.

   UPDATE uses the control file (ctlfile CNTRL) to update the assembler language source file. If updates exist, the new file is named $fn ASSEMBLE. Otherwise, it is called fn ASSEMBLE.

   UPDATE stacks information from the control file in the console stack, and prints the update log file.

   If a PTF file is missing, a message stating this will be written to both the console and to the update log file.

2. Using the library list from the MACS record in the control file, VMFASM issues a GLOBAL MACLIB command.

3. The updated source file, $fn ASSEMBLE, is assembled using the options indicated on the VMFASM command line.

4. The output text deck from the assembly, $fn TEXT, is concatenated with the UPDATES file so that the text deck contains a history of update activity.

5. Using the update level identifier from the control file (the identifier of the most recent update that was found and applied is stacked by the UPDATE command), VMFASM determines how to rename $fn TEXT.

   If the update level identifier is TEXT, the text deck is renamed fn TEXT.

   If the update level identifier is anything other than TEXT, the text deck is renamed fn TXTxxxxx (where xxxxx is the 1 to 5 character update level identifier). The new fn TEXT (or TXTxxxxx) is found on the 'A' disk.

6. Temporary files ($fn ASSEMBLE or fn ASSEMBLE, fn UPDATES, and fn CTLFILE are erased.

A summary of the input and output files used by VMFASM follows.

DISK INPUT FILES:

fn ASSEMBLE	Assembler Language source
ctlfile CNTRL	control file

MACLIBS, auxiliary control files (AUXxxxxx), and miscellaneous update files.

DISK OUTPUT FILES:

fn {TEXT {TXTxxxxx}	object deck, named according to the update level identifier in the control file

This file also contains data from the UPDATES file, together with date and time information.

PRINTER OUTPUT FILES

fn ctlfile	Assembler listing (if PRINT options is in effect)

This file also contains data from the update log file (fn UPDLOG), describing the updates applied to the source file.

Responses

The UPDATE command issues the message DMSUPD178I to indicate each of the update files being applied.

fn TEXT A HAS BEEN ERASED.

indicates that the text file to be produced does not have a filename of TEXT. File 'fn TEXT A' already existed. This prevents you from inadvertently including a down-level textfile in your nucleus.

ASMBLING fn

indicates that the assembly is going to begin. If you specified any assembler options on the VMFASM command line, the options used are also displayed.

fn {TEXT } A1 CREATED
{TXTxxxxx}

indicates the filename and filetype of the text deck.

Error Messages

```
DMSSTT002E FILE 'fn ASSEMBLE *' NOT FOUND
     The source file could not be located.

DMSSTT002E FILE 'ctlfile CNTRL *' NOT FOUND
     The control file could not be located.

*** ERROR UPDATING fn ***
     A return code from the UPDATE command was greater than
     12 but not equal to 40.  This means that a severe error
     occurred.  VMFASM terminates, and the return code
     from UPDATE is returned.

*** ERROR ASMBLING fn ***
     An assembler error occurred.

*** fn {TEXT    } A1 IS NOT CREATED ***
       {TXTxxxxx}
     No text file was produced because of assembler errors.
     VMFASM terminates with the return code from the ASSEMBLE
     command.

fn {TEXT    } A WAS NOT CREATED, RC=rc FROM COPYFILE
   {TXTxxxxx}
     The COPYFILE command failed to merge the fn UPDATES A1
     files or your A-disk is full.  The text file, therefore,
     failed with the return code shown.  VMFASM terminates
     with the return code from the COPYFILE command.
```

VMFDOS

Use of the VMFDOS command is restricted to creating CMS files containing VSE modules for specific installation purposes only. VMFDOS uses either a VSE distribution library tape or VSE SYSIN tape to install only DOS/VS RPG II and VSE/VSAM Program Products.

The format of the VMFDOS command is:

```
VMFDOS      ┌      ┐ ┌ ┌   ┐ ┐ ┌        ┐
            │ LOAD │ │ │181│ │ │PRIVate │  [fn] [(options...)]
            │ SCAN │ │ │182│ │ │SYSTem  │
            └      ┘ │ └   ┘ │ │SYSIN   │
                     │ ┌    ┐│ └        ┘
                     │ │TAP1││
                     │ │TAP2││
                     │ └    ┘│
                     └       ┘
            PRIV or SYST options:

                    CSL  ┌*           ┐
                         │xxx*        │
                         └module name ┘

                    RL   ┌*           ┐
                         │yyy*        │
                         └module name ┘

                    SL   ┌*           ┐
                         │zzz*        │
                         └module name ┘

            SYSIN options:

                  {ALL   }
                  {SELect}
```

where:

LOAD
: loads the specified VSE modules as CMS files from the VSE tape. The modules to be loaded are specified:

- Via the SL or RL options.
- Via a scan file. (See Usage Note 7.)

LOAD is the default function.

SCAN
: specifies that the tape is to be scanned and a file created that contains the names of all the VSE modules. This file is ‘fn SCAN A1’ if fn is specified or ‘VMFDOS SCAN A1’. Any options specified are ignored.

TAP1 or 181
TAP2 or 182
: specifies the symbolic tape identification (TAPn) or the actual device address (18n) of the tape. TAP1 or 181 is the default.

PRIVate specifies that the VSE private libraries are to be processed. This is the default parameter.

SYSTem specifies that the VSE system libraries are to be processed.

SYSIN specifies that the VSE tape is in SYSIN format.

fn specifies the name of a CMS file to be used with either the load or scan functions. The filetype must be 'SCAN'. The load function reads the file to determine which VSE modules are to be loaded as CMS files. The scan function creates (in a file called 'fn SCAN') all the VSE module names contained on the tape.

PRIV or SYST Options:

CSL specified without an option is valid only if a scan filename has also been specified. All of the macros listed in the scan file are loaded and decompressed.

* specifies that all the macros on the VSE/VSAM Optional Source Statement Library tape are to be read in and decompressed into CMS files.

xxx* specifies that all VSE modules beginning with the characters xxx are to be processed.

module name
is a specific module name.

Note: The CSL option is only supported for use on the VSE/VSAM Optional Source Statement Library.

RL * specifies that all the VSE modules in the relocatable library are to be processed. This is the default if no options are specified, if fn is not specified, and if 'VMFDOS SCAN' does not exist.

yyy* specifies that all VSE modules beginning with the characters yyy are to be processed.

module name
is a specific module name.

SL * specifies that all the VSE modules in the source statement library are to be processed.

zzz* specifies that all VSE modules beginning with the characters zzz are to be processed.

module name
is a specific module name.

SYSIN Options:

ALL specifies that all VSE modules found are to be loaded as CMS files.

SELect indicates that you will be prompted prior to writing each VSE module to disk. You must indicate whether or not each module is to be included.

Examples

The following are examples of invoking the VMFDOS command.

```
vmfdos (rl abc*
```

All VSE modules in the private relocatable library that begin with 'ABC' are loaded as CMS files. The CMS files created have a filetype of 'TEXT'; they have a filename that is the same as the VSE module name.

```
vmfdos syst (rl abc*
```

All VSE modules in the system relocatable library that begin with 'ABC' are loaded as CMS files. The CMS files created have a filetype of 'TEXT'; they have a filename that is the same as the VSE module name.

```
vmfdos (sl xyz1*
```

All VSE modules in the private source statement library and all the private relocatable library modules that begin with 'XYZ1' are loaded as CMS files. The CMS files created have a filetype of 'MACRO'; they have a filename that is the same as the VSE module name.

```
vmfdos syst (sl *
```

All VSE modules in the system source statement library and the system relocatable library are loaded as CMS files. The CMS files created have a filetype of 'MACRO'; they have a filename that is the same as the VSE module name.

```
vmfdos priv (rl abc* sl *
```

All VSE modules that begin with 'ABC' in the private relocatable library and all modules in the private source library are loaded as CMS files. The CMS files have a filename the same as the VSE module name and a filetype of 'TEXT' or 'MACRO', as appropriate.

```
vmfdos scan syst
```

The VSE system libraries are scanned. File 'VMFDOS SCAN' is created to contain the names of the VSE modules.

```
vmfdos scan sysin fn
```

The VSE SYSIN tape is scanned. File 'fn SCAN' is created to contain the names of the VSE modules.

```
vmfdos
```

All VSE modules in the private relocatable libraries are loaded as CMS files. The CMS files created have a filename the same as the VSE module name and a filetype of 'TEXT'.

```
vmfdos sysin fn
```

File 'fn SCAN' contains a list of module names. The modules are on the VSE SYSIN tape and are loaded as CMS files. The CMS files created have a filename the same as the VSE module name and a filetype of 'TEXT' or 'MACRO', as appropriate.

```
vmfdos sysin
```

File 'VMFDOS SCAN' contains a list of module names. These modules are on the VSE SYSIN tape and are loaded as CMS files. The CMS files created have a filename the same as the VSE module name and a filetype of 'TEXT' or 'MACRO', as appropriate.

Usage Notes

1. When the scan function is invoked, VMFDOS erases either the CMS file 'fn SCAN A1' or 'VMFDOS SCAN A1' prior to scanning the tape.

2. The tape must be positioned at the file to be processed (via the TAPE FSF command) prior to issuing the VMFDOS command.

3. VMFDOS erases those existing files that have the same filename as a VSE module being loaded.

4. VMFDOS uses the CMS 'A' disk to write all files. A message is issued if a R/W 'A' disk is not available.

5. VSE modules loaded from a relocatable library tape file will have a filetype of 'TEXT' when written as a CMS file.

6. VSE modules loaded from a source statement library tape file will have a filetype of 'MACRO' when written as a CMS file.

7. If a filename is specified when using the LOAD function, any options specified are ignored and the 'fn SCAN' file is used. If a filename is not specified, but a 'VMFDOS SCAN' file exists, any options specified are ignored and the 'VMFDOS SCAN' file is used.

8. VMFDOS can only handle 430 entry requests at a time when using a 'fn SCAN' or 'VMFDOS SCAN' file as input. You can load more than 430 modules by using generic names in the scan file or separate scan files.

Error Messages

```
DMSDOS787E ERROR DECOMPRESSING macro
           MACRO
ONLY THE FIRST 430 ENTRIES LISTED WILL BE PROCESSED, BUFFER
IS FULL
```

An informational message only; processing continues. VMFDOS only acquires enough storage to contain 430 entry requests when a filename is specified or 'VMFDOS SCAN' exists and is used as input.

VMFLOAD

Use the VMFLOAD command to generate a new CP or CMS nucleus. The VMFLOAD program uses two files, a loadlist EXEC file and a control file, to produce a punch file that has several object modules. VMFLOAD requires a virtual machine with 512K.

The format of the VMFLOAD command is:

VMFLOAD	loadlist ctlfile

where:

`loadlist` is the filename of an EXEC file that contains the names of object modules in the order in which they are to reside in the complete load file for the nucleus. For example:

```
&CONTROL OFF
&1 &2 fn [ft]
&1 &2 fn [ft]
  .
  .
  .
```

where fn and optionally ft, are the filename and filetype of an object module to be punched. The object modules are punched in the order specified, beginning at the top of the loadlist EXEC. If a filetype is specified, VMFLOAD searches for that specific file, and, if it finds it, punches it without a header card.

If the filetype is not specified in the loadlist, VMFLOAD uses the control file to determine which object module is the highest level object module available. VMFLOAD searches the control file from top to bottom. When it finds the appropriate object module, VMFLOAD punches it.

`ctlfile` is the filename of the control file. This is usually the same control file used to apply updates to modules via the VMFASM or UPDATE commands. This file identifies the highest level object module available, if the filetype is not specified in the loadlist.

How VMFLOAD Works

1. Before the the files specified in the loadlist are punched, VMFLOAD issues a SPOOL PUN CONT command to ensure that the punched files appear as one deck. You may wish to specify SPOOL PUN TO 'userid' before you invoke VMFLOAD to transfer the punched output as a file to your own (or another) virtual machine as a reader file. If you want to perform any additional controls, you should write an EXEC procedure to perform the control and invoke VMFLOAD from that EXEC procedure.

2. For each entry in the loadlist that does not specify a filetype VMFLOAD searches the control file to determine the filetype of the object module. The filetypes are based on the update level identifiers in the control file. These are the identifiers used by the VMFASM to assign filetypes to object decks.

Remember that updates applied to source files are applied from the bottom of the file towards the top. Therefore, VMFLOAD searches the control file from the top towards the bottom to locate the most recent update level.

For example, if a control file contains the records and control file, you would issue:

```
TEXT MACS DMKMAC
LOCAL FIX1
SPEC  AUX1111
PTF   C12765DK
IBM1  AUXSP
```

Then for each entry in the loadlist, the VMFLOAD search order is:

- fn TXTLOCAL
- fn TXTSPEC
- fn TXTIBM1

As soon as VMFLOAD locates a file, it punches it, then continues processing the next entry in the loadlist. If none of the above filetypes exist for the loadlist entry, VMFLOAD searches for filename TEXT. If there is no TEXT file, VMFLOAD displays a message and continues processing with the next entry in the loadlist.

Note: VMFLOAD ignores records that have an update level identifier of PTF, and so searches for the next lowest level identifier when determining the filetypes of object modules to punch.

3. When all the object modules are punched, VMFLOAD issues the commands

```
SPOOL PUNCH NOCONT
CLOSE PUNCH
```

If you issued the command

```
spool punch to *
```

prior to invoking VMFLOAD, the completed load deck is placed in your virtual card reader.

DISK INPUT FILES:

loadlist EXEC — contains the filenames, and optionally filetypes, of the object modules to be punched.

DMKLD00E LOADER — the loader, which should be the first entry in the loadlist EXEC.

object modules with filetypes of TEXT or TXTxxxxx, where xxxxx is the update level identifier in a control file, used by VMFASM to name the object module.

PUNCH OUTPUT FILES:

load deck	punched to your virtual machine

Notes:

1. The distributed system uses the following loadlists:

Loadlist		**Usage**
APLOAD	EXEC	CP nucleus without V=R for an AP or MP system
AVLOAD	EXEC	CP nucleus with V=R for an AP or MP system
CMSLOAD	EXEC	CMS nucleus
CMSLOADL	EXEC	CMSL nucleus
CPLOAD	EXEC	CP nucleus without V=R for UP
CPLOADSM	EXEC	CP nucleus without V=R for UP using Small CP option
VRLOAD	EXEC	CP nucleus with V=R for UP

   For example, to punch a new CP nucleus with the distributed loadlist, enter:

   ```
   vmfload cpload dmksp
   ```

   The GENERATE and VMSERV EXECs use VMFLOAD to generate a new CP nucleus.

2. After you have punched a new nucleus with VMFLOAD, you can either move the nucleus to tape, using the MOVEFILE command, or, if the nucleus is in your virtual card reader, you can IPL it:

   ```
   ipl 00c
   ```

   When you IPL the virtual card reader, the loader is read first, and it loads the rest of the object modules. If the loader is successful, the nucleus is written on disk, and the load map is spooled to the virtual printer. If you want to preserve a disk copy of the load map, you should spool your printer to your virtual card reader, then read the file onto disk.

3. To locate file(s), VMFLOAD searches all of your accessed disks using the standard search order, A through Z.

Responses

```
SYSTEM LOAD DECK COMPLETE
```

This message is displayed when all the files in the loadlist have been punched.

Error Messages

```
INSUFFICIENT OR INVALID ARGUMENTS
     The command line waaaaas incorrectly entered.

NO CONTROL FILE
     The control file could not be located.

ERROR IN CONTROL FILE
     The control file contains an invalid record.

NO LOAD LIST
     The loadlist could not be located.
ERROR IN LOAD LIST
     The loadlist contains an invalid record.

fn ft NOT FOUND
     No text file was found.

ERROR ON PUNCH
     An error occurred punching a file.
```

CP Loadlist Requirements

The CPLOAD loadlist EXEC contains a list of CP modules that is used by the VMFLOAD procedures to punch the text decks for the CP system. All modules following DMKCPE in the list are pageable CP modules. Each 4K page in this area may contain one or more modules. Pageable modules must not span the 4K page boundaries. The module grouping is governed by SPB (Set Page Boundary) cards. An SPB card is a loader control card that forces the loader to start this module at the next higher 4K boundary. If more than one module is to be contained in a 4K page, only the first is preceded by an SPB card.

The loader inserts SPB cards automatically where they are needed; you need not insert SPB cards.

The position of two modules in the loadlist is critical. All modules following DMKCPE must be reenterable and must not contain any address constants referring to anything in the pageable CP area. DMKCKP must be the last module in the loadlist.

The last entry in a CP loadlist must be:

```
&1  &2  &3  LTD  DMKSAVNC
```

<table>
<tr><td>X'02'</td><td>LDT | not used | DMKSAVNC | blank</td></tr>
</table>

For more information, see the Loader Terminate Statement section in the *VM/SP CMS Command and Macro Reference.*

For V=R users only, DMKSLC is required. It is generated in either of two ways:

1. GENERATE, which will automatically prompt the user, or

2. Use VRSIZE to generate DMKSLC and then issue VMFLOAD. (See the flowchart, Figure 17 on page 152. "Deciding Which Updating Procedures to Use.")

VMFMAC

Use the VMFMAC EXEC procedure to update macro libraries. It invokes the CMS UPDATE command to update specified copy or macro files, according to entries in a control file, and then builds a new macro library from the resulting new versions of those files.

Notes:

1. When files with MACRO filetypes are added to a MACLIB, the membername is taken from macro prototype statement. When files with COPY filetypes are added to a MACLIB, the membername is taken from the filename of the COPY file. This will be $filename if updates were found, otherwise it is just the filename. If you include a *COPY statement as the first record in the file, in the format:

   ```
   *COPY membername
   ```

   the MACLIB directory uses membername to name the copy file.

2. If errors occur during VMFMAC processing, consult the NEWMAC COPY file printed by VMFMAC. If you can correct the errors involving one or two macro or copy files, add these members to NEWMAC MACLIB using the MACLIB command. Then, enter the following commands:

   ```
   erase libname maclib
   rename newmac maclib fm libname = =
   ```

   The current libname MACLIB is erased and NEWMAC MACLIB is renamed libname MACLIB.

The format of the VMFMAC command is:

VMFMAC	libname ctlfile

where:

`libname` is the filename of the macro library to be updated, and of the EXEC file that contains the names of the library members. The entries in libname EXEC must be in the following format:

```
&1 &2 fn1
&1 &2 fn2
  .
  .
  .
```

where fn1, fn2, and so on, are filenames of macro or copy files to be updated and included in the macro library, which must have a filetype of MACLIB.

`ctlfile` is the filename of a control file to be used to apply the updates. The filetype must be CNTRL. The filenames used by VM/SP are DMKSP, DMKSPA, DMKSPM, DMSSP, and DMSMSP.

How VMFMAC Works

The steps taken by VMFMAC are summarized below.

1. VMFMAC locates libname EXEC and the control file. It also erases any existing files named NEWMAC MACLIB and NEWMAC COPY. Then VMFMAC begins reading the macro or copy filenames from the EXEC, beginning at the bottom.

2. For each entry in the libname EXEC, VMFMAC:

   - Invokes the UPDATE command with the CTL option to apply the updates specified in the control file.

   - Adds the updated macro or copy file ($filename MACRO or $filename COPY) to the macro library NEWMAC MACLIB.

   - Adds the UPDATES file created by the UPDATE command to the file NEWMAC COPY.

   - Erases $filename MACRO or $filename COPY, and filename UPDATES.

3. If there are no update files for a macro or copy file specified in libname EXEC, the macro or copy file is added to NEWMAC MACLIB in its current form. NEWMAC COPY, containing a history of updates applied by VMFMAC, is added to NEWMAC MACLIB.

4. If no errors occur during the procedure, then when all the macros have been added to NEWMAC MACLIB, NEWMAC MACLIB is renamed libname MACLIB. libname MACLIB, if it exists, is erased.

   If errors occur during the VMFMAC EXEC procedure (for example, if a MACRO or a COPY file is not found) libname MACLIB is not erased, and the updated macro library retains the name NEWMAC MACLIB.

DISK INPUT FILES

libname EXEC contains a list of macro a copy file to be updated and/or included in libname MACLIB.

ctlfile CNTRL is the control file used by the UPDATE command.

MACRO and COPY files to be updated and/or included in the macro library, plus miscellaneous auxiliary control files and update files.

DISK OUTPUT FILES

libname MACLIB is the updated macro library.

libname COPY contains the UPDATES files produced by UPDATE command processing.

PRINTER OUTPUT FILES

The printer is spooled with the CONT option, so that when VMFMAC completes, the printer file contains:

- A copy of the control files.
- For each updated macro or copy file, the update log file produced by the UPDATE command.
- A copy of each macro or copy file in the macro library.
- The libname COPY file, which contains the accumulated UPDATES files created by the UPDATE command.

Responses

The UPDATE command issues the message DMSUPD178I to inform you of the updates being applied to each macro or copy file. If no updates are found, message DMSUPD181E is issued.

fname $\left\{\begin{matrix}\text{COPY}\\ \text{MACRO}\end{matrix}\right\}$ ADDED.
indicates that the specified macro or copy file has been added to the macro library.

libname COPY ADDED.
indicates that libname COPY, containing the update history, of the MACLIB, has been added.

Error Messages

```
*** TYPE 'VMFMAC LIBNAME CTL' ***
```

This message indicates that the command line did not have two operands.

```
*** libname EXEC NOT FOUND ***
```

VMFMAC could not locate the EXEC file associated with the macro library.

```
*** ctlfile CNTRL NOT FOUND ***
```

VMFMAC could not locate the control file.

```
*** fn COPY or MACRO NOT FOUND ***
```

A library member named in libname EXEC could not be located.

```
*** ERRORS UPDATING fn {COPY } ***
                       {MACRO}
    fname {COPY } NOT INCLUDED IN MACLIB
          {MACRO}
```

This message indicates an UPDATE command error occurred for the member, and the file was not written into the MACLIB.

```
DUE TO PREVIOUS ERRORS, THE RESULT OF THIS MACLIB BUILD
IS CALLED 'NEWMAC MACLIB', libname MACLIB HAS
NOT BEEN REPLACED
```

One or more errors were encountered, and you must correct them and create the MACLIB yourself.

VMFPLC2

Use the VMFPLC2 command to:

- Load source code from the Product Tape
- Load the service installation VMSERV EXEC from the Program Update Tape
- Dump CMS-formatted files from disk to tape. (Disk files to be dumped can contain either fixed-length or variable-length records.)
- Load previously dumped files from tape to disk
- Perform various control operations on a specified tape drive.

Note: The VMFPLC2 command does not process multi-volume files. Files processed by the VMFPLC2 command must be CMS-formatted.

The format of the VMFPLC2 command is:

```
VMFPLC2   ⎧ DUMP    {fn} {ft} [fm]        [(optionA optionB  ]⎫
          ⎪         {* } {* } [* ]        [  optionD)        ]⎪
          ⎪                                                   ⎪
          ⎪ LOAD   [{fn} {ft} [fm]]       [(optionB optionC  ]⎪
          ⎪        [{* } {* } [A ]]       [ optionD optionE) ]⎪
          ⎪                                                   ⎪
          ⎨ SCAN   [{fn} {ft}]            [(optionB optionC  ]⎬
          ⎪        [{* } {* }]            [  optionF)        ]⎪
          ⎪                                                   ⎪
          ⎪ SKIP   [{fn} {ft}]            [(optionB optionC  ]⎪
          ⎪        [{* } {* }]            [  optionD)        ]⎪
          ⎪                                                   ⎪
          ⎪ MODESET            [(optionD)]                    ⎪
          ⎪ tapcmd   [n]       [(optionD)]                    ⎪
          ⎩          [1]                                      ⎭

          optionA:   [WTM   ]
                     [NOWTM ]

          optionB:   [NOPRint]
                     [PRint  ]
                     [Term   ]
                     [DISK   ]
                     [APPend ]

          optionC:   [EOT    ]
                     [EOF n  ]
                     [EOF 1  ]

          optionD:   [ cuu ]      [DEN den]
                     [ 181 ]

          optionE:   [SELect]  [STOP]

          optionF:   [DATE]
```

where:

```
DUMP {fn}{ft}[fm]
     {* }{* }[* ]
```

dumps one or more disk files to tape. If fn and/or ft is specified as an asterisk (*) all files that satisfy the other file identifier are dumped.

If fm is coded as a letter, that disk and its extensions are searched for the specified file(s). If fm is coded as a letter and number, only files with that mode number and letter (and the extensions of the disk referenced by that fm letter) are dumped. If fm is coded as asterisk (*), all accessed disks are searched for the specified file(s). If fm is not specified, only the A-disk and its extensions are searched.

```
LOAD  ┌┌fn┐┌ft┐┌fm┐┐
      │{  }{  }│  ││
      └└* ┘└* ┘└A ┘┘
```

reads tape files onto disk. If a file identifier is specified, only that one file is loaded. If the option EOF n is specified and no file identifier is entered, n tape files are written to disk. If an asterisk (*) is specified for fn or ft, all files within EOF n that satisfy the other file identifier are loaded.

The files are written to the disk indicated by the filemode letter. The filemode number, if entered, indicates that only files with that filemode number are to be loaded.

```
SCAN  ┌┌fn┐┌ft┐┐
      │{  }{  }│
      └└* ┘└* ┘┘
```

positions the tape at a specified point, and lists the identifiers of the files it scans. Scanning occurs over n tape marks, as specified by the option EOF n (the default is 1 tape file). However, if a file identifier (fn and ft) is specified, scanning stops upon encountering that file; the tape remains positioned ahead of the file.

```
SKIP  ┌┌fn┐┌ft┐┐
      │{  }{  }│
      └└* ┘└* ┘┘
```

positions the tape at a specified point and lists the identifiers of the files it skips. Skipping occurs over n tape marks, as specified by the option EOF n (the default is 1 tape mark). However, if a file identifier (fn and ft) is specified, skipping stops after encountering that file; the tape remains positioned immediately following the file.

MODESET sets the values specified by the DEN, TRACK, and TRTCH options. After initial specification in a VMFPLC2 command, these values remain in effect for the virtual tape device until they are changed in a subsequent VMFPLC2 command, RDTAPE, WRTAPE, or TAPECTL macro.

```
tapcmd┌ ┐
      │n│
      │1│
      │-│
      └ ┘
```

specifies a tape control function (tapcmd) to be executed n times. Default is 1 if n is not specified. These functions also work on tapes in a non-CMS format.

Tapecmd	Action
BSF	Backspace n tape marks
BSR	Backspace n tape records
ERG	Erase gap
FSF	Forward-space n tape marks
FSR	Forward-space n tape records
REW	Rewind tape to load point
RUN	Rewind tape and unload
WTM	Write n tape marks

Options:

If conflicting options are specified, the last one entered is in effect.

WTM writes a tape mark on the tape after each file is dumped.

NOWTM writes a tape mark after each file is dumped, then backspaces over the tape mark so that subsequent files written on the tape are not separated by tape marks.

NOPRINT does not spool the list of files dumped, loaded, scanned, or skipped to the printer.

PRINT spools the list of files dumped, loaded, scanned, or skipped to the printer.

TERM displays a list of files dumped, loaded, scanned, or skipped at the terminal.

DISK creates a disk file containing the list of files dumped, loaded, scanned, or skipped. The disk file has the file identification of TAPE MAP A5.

APPEND causes the disk file (containing the list of files dumped, loaded, scanned, or skipped), to be added to the end of an existing TAPE MAP.

EOT reads the tape until an end-of-tape indication is received.

EOF n reads the tape through a maximum of *n* tape marks. The default is EOF 1.

cuu
18n specifies the virtual device address of the tape to be read or written to where n is 1, 2, 3, or 4. The default is 181. The unit specified by cuu must previously have been attached to your CMS virtual machine before any tape I/O operation can be attempted. Only virtual device addresses 181 through 184 are supported.

DEN den is the tape density where den is 200, 556, 800, 1600, or 6250. If 200 or 556 is specified, 7TRACK is assumed. If 1600 or 6250 is specified, 9TRACK is assumed; if 800 is specified, 9TRACK is assumed unless 7TRACK is specified. In the case of either 800/1600 or 1600/6250 dual-density drives, 1600 is the default if the 9TRACK option is specified. If neither the 9TRACK option nor the DEN option is specified, the drive operates at whatever bpi the tape drive was last set.

SELECT inhibits loading of a file from the tape that causes replacement of an *identical* file on the disk. Files will be loaded only if they do not exist on the specified disk, or when the date/time stamp for the file on the disk does *not match* the date/time stamp for the corresponding file on the tape.

STOP assumes that files contained on the tape are in alphabetical sequence. If the requested file is on the tape, the file is loaded onto disk and the tape stops. If the file is not on the tape and a file is encountered that is alphabetically beyond the bounds of the requested file, the tape stops. You must specify 'fn ft'. Neither 'fn' nor 'ft' may be specified as '*'.

DATE displays listfile information during a SCAN. The information displayed includes number of records, length of records, and date/time stamp.

Usage Notes

1. Tape records written by the VMFPLC2 command are 4005 bytes long. The first character is a binary 2 (X'02'), followed by the characters CMS and a file format byte, followed by 4000 bytes of file data packed without regard for logical record length. If a null block is dumped, the character "0" replaces the byte after CMS. This causes subsequent loading of null blocks to be ignored. In the final record, the character N replaces the blank after CMS, and the data area contains CMS file directory information.

2. If a tape file contains more CMS files than would fit on a disk, the tape load operation may terminate if there is not enough disk space to hold the files. To prevent this, when you dump the files, separate logical files by tape marks, then forward space to the appropriate file.

3. The CMS file directory is the first record of each CMS file on tape.

4. It is possible to run a tape off the reel in at least one situation. If you specify EOF *n* and *n* is greater than the number of tape marks on the tape, the tape will run off the reel.

5. The options for the 8809 tape drive must be 9TRACK and DEN 1600. Note that these are the default values, so you do not need to specify them.

6. For more information on tape file handling, see the *VM/SP CMS User's Guide.*

VRSIZE

Use the VRSIZE command to build a textfile for generating a virtual=real area in your system.

The format of the VRSIZE command is:

VRSIZE	

If you build your nucleus with your own build EXEC and want to have a V=R area in your system, a DMKSLC textfile needs to be generated.

VRSIZE will build a file called DMKSLC TEXT. You use the VRSIZE command to generate DMKSLC provided a loadlist that lists DMKSLC is used. DMKSLC comes right after DMKPSA (page zero) in the CP nucleus. It moves the set location counter from the beginning of page one to the end of the V=R area. Virtual=Real size must be specified in thousands (K), rather than millions (M). For example: 8192K instead of 8M. This only needs to be done once, as long as you don't want to change your V=R size. Several different DMKSLC textfiles may be kept around for different purposes.

To create textfile DMKSLC, enter the VRSIZE command and reply to the prompts, as in the example that follows:

```
vrsize

VIRTUAL=REAL OPTION REQUIRED (YES,NO):

yes

STORAGE SIZE OF VIRT=REAL  (MINIMUM IS 32K):

5120k (Not 5m)            Specify your own V=R size here.

05120K STORAGE SIZE FOR VIRTUAL=REAL
IS THE ABOVE ENTRY CORRECT (YES,NO)

yes

R;

l dmkslc ( d

FILENAME FILETYPE FM FORMAT LRECL RECS BLOCKS DATE     TIME
DMKSLC   TEXT     A1 F      80    3    1      mm/dd/yy hh:mm:ss
```

VSEVSAM

Use the VSEVSAM EXEC to obtain VSE/VSAM Assembler Language macros from the Licensed Optimal Machine Readable Materials tape. The VSEVSAM EXEC creates the VSEVSAM MACLIB for you. Once the maclib is created, it contains all of the VSE/VSAM assembler language macros, and the following VSE macros: CDLOAD, CLOSE, CLOSER, GET, OPEN, OPENR, and PUT.

The format of the VSEVSAM EXEC is:

VSEVSAM	

Example Using VSEVSAM

Before invoking the VSEVSAM EXEC, complete the following:

1. Mount the Licensed Optimal Machine Readable Materials tape at virtual address 181.

2. Load the seven VSE macros from the Product Tape to MAINT 393 or a minidisk of your choice. (As long as the macros are available when VSEVSAM is invoked, the actual minidisk used is not critical.)

   Note: Because the seven VSE macros: CDLOAD, CLOSE, CLOSER, GET, OPEN, OPENR, and PUT will be loaded in the MACLIB, they can be erased from the disk after the MACLIB is created.

To invoke VSEVSAM EXEC, enter:

```
  vsevsam
```

Once invoked, VSEVSAM EXEC prompts you for information. For example, the system responds:

```
DMSWVV797I "QUIT" MAY BE ENTERED IN RESPONSE TO ANY QUERY TO END
           PROCESSING.

DMSWVV788R ARE THE MACROS TO BE READ FROM TAPE OR ARE
           THEY ALREADY ON DISK?  REPLY (TAPE | DISK).
           IF A DEFAULT OF TAPE IS TO BE USED, PRESS "ENTER."
```

For this example we will read the macros from disk. Enter:

```
disk
```

The system responds:

```
DMSWVV790R IF THE DEFAULT LIBRARY NAME OF "VSEVSAM" IS TO BE
           USED, PRESS "ENTER."  ELSE, ENTER THE NAME TO BE USED
           FOR THE LIBRARY.
```

We want to call the library 'vsammacs', so enter:

```
vsammacs
```

The system responds:

```
DMSWVV791I THE LIBRARY NAME WILL BE "VSAMMACS."
           PRESS "ENTER" TO CONTINUE, ELSE ENTER
           "QUIT" OR THE NAME TO BE USED FOR THE LIBRARY.
```

Because 'vsammacs' is the name we want to call the library, press the ENTER key to continue.

The system responds:

```
DMSWVV808R MACRO LIBRARY "VSAMMACS" WILL BE ERASED.
           PRESS "ENTER" TO CONTINUE OR "QUIT" TO EXIT.
```

Because we want to erase old versions of the library that might be on our A-disk, press the ENTER key.

The system responds:

```
DMSWVV793I MACLIB GENERATION COMPLETED.

DMSWVV792R ARE THE MACROS TO BE ERASED FROM DISK?
           REPLY (YES | NO).  PRESS "ENTER" FOR DEFAULT OF "YES."
```

We want to erase the macros from disk. Press the ENTER key.

The system responds:

```
  DMSWVV802I MACROS ERASED - VSEVSAM PROCESSING COMPLETE.
  R;
```

Usage Note:

When you want to obtain the VSE/VSAM assembler language macros, before executing the VSEVSAM EXEC, mount the Licensed Optimal Machine Readable Materials tape on your virtual 181 tape drive.

Responses

```
DMSWVV788R ARE THE MACROS TO BE READ FROM TAPE OR ARE THEY
           ALREADY ON DISK?  REPLY (TAPE | DISK).  IF A DEFAULT
           OF TAPE IS TO BE USED, PRESS "ENTER."
DMSWVV790R IF THE DEFAULT LIBRARY NAME OF "VSEVSAM" IS TO BE
           USED, PRESS "ENTER."  ELSE, ENTER THE NAME TO BE USED
           FOR THE LIBRARY.
DMSWVV791I THE LIBRARY NAME WILL BE "libname."
           PRESS "ENTER" TO CONTINUE, ELSE ENTER
           "QUIT" OR THE NAME TO BE USED FOR THE LIBRARY.
DMSWVV792R ARE THE MACROS TO BE ERASED FROM DISK?
           REPLY (YES | NO).  PRESS "ENTER" FOR DEFAULT OF "YES."
DMSWVV797I "QUIT" MAY BE ENTERED IN RESPONSE TO ANY QUERY TO END
           PROCESSING.
DMSWVV798R THE VSE/VSAM OPTIONAL SOURCE STATEMENT LIBRARY TAPE
           MUST BE MOUNTED AS VIRTUAL 181.  IF IT IS NOT, ENTER
           "QUIT" HERE AND HAVE THE TAPE MOUNTED.  ELSE, PRESS
           "ENTER" TO CONTINUE.
DMSWVV801I ARGUMENTS ENTERED ARE IGNORED.
DMSWVV802I MACROS ERASED - VSEVSAM PROCESSING COMPLETE.
DMSWVV808R MACRO LIBRARY "VSAMMACS" WILL BE ERASED.
           PRESS "ENTER" TO CONTINUE OR "QUIT" TO EXIT.
```

Other Messages and Codes

```
DMSWVV789W INVALID RESPONSE.
DMSWVV794E ERROR IN MACLIB GENERATION.
DMSWVV795E ERROR READING MACROS FROM TAPE.
DMSWVV796E ERROR READING FROM "VSEVSAM SCAN" FILE.
DMSWVV799E ERROR READING FROM "VSEVSAM SCAN" FILE - ALL MACROS
           MAY NOT BE ERASED.
DMSWVV800E ONE OF THE FILES NEEDED FOR MACLIB GENERATION IS
           MISSING.
DMSWVV809E ERROR COPYING "VSEVSAM SCAN" FILE FROM S-DISK TO
           A-DISK.
```

Appendixes

Sample file listings are in Appendixes J through P.

Appendix A. CP/CMS Nucleus/Module/Segment Regeneration Requirements

Use this appendix when applying updates and local fixes. When corrective service is shipped via PUT, all modules are regenerated and replaced on a one-to-one basis. Whenever CMS source code is updated, the CMS nucleus and/or some CMS modules/segments must be regenerated. The following table shows which must be regenerated in each case. (If a source name does not appear in the table, the file is contained within the CMS nucleus.)

Change in Source	Requires Regeneration of Module/Segment	EXEC Procedure To Use
DMSACC	ACCESS, Nucleus	CMSGEND
DMSACF	ACCESS, Nucleus	CMSGEND
DMSACM	ACCESS, Nucleus	CMSGEND
DMSAMS	AMSERV	CMSGEND
DMSARE	RELEASE	CMSGEND
DMSASD	ASSEMBLE	CMSGEND
DMSASM	ASSEMBLE	CMSGEND
DMSASN	ASSGN	CMSGEND
DMSBAB	CMSDOS	DOSGEN
DMSBOP	CMSDOS	DOSGEN
DMSBTB	CMSBATCH	CMSGEND
DMSBTP[1]		
DMSCCK	CATCHECK	CMSGEND
DMSCLS	CMSDOS	DOSGEN
DMSCMP	COMPARE	CMSGEND
DMSCPY	COPYFILE	CMSGEND
DMSCVH	CMSDOS	DOSGEN
DMSDAS	CMSDOS	DOSGEN
DMSDLB	DLBL	CMSGEND
DMSDLK	DOSLKED	CMSGEND
DMSDMP	CMSDOS	DOSGEN
DMSDOS	CMSDOS	DOSGEN
DMSDSK	DISK	CMSGEND
DMSDSL	DOSLIB	CMSGEND
DMSDSV	DSERV	CMSGEND
DMSEDC	EDIT[2]	CMSGEND
DMSEDF	EDIT[2]	CMSGEND
DMSEDI	EDIT[2]	CMSGEND
DMSEDX	EDIT[2]	CMSGEND
DMSEND	CMSAMS	VSAMGEN (AMS)
DMSETR	CMSDOS	DOSGEN
DMSFCH	CMSDOS	DOSGEN
DMSFLD	FILEDEF[3]	CMSGEND
DMSFLE	FILEDEF[3]	CMSGEND
DMSFOR	FORMAT	CMSGEND
DMSGIO	EDIT[2]	CMSGEND

[1] DMSBTP is loaded into free storage by DMSBTB. The file DMSBTB TEXT must be filemode 2 on the system disk.

[2] When the CMSGEND EXEC procedure is invoked for EDIT, it creates the EDIT module. Then, it automatically reinvokes itself to create the EDMAIN module.

[3] When the CMSGEND EXEC is invoked for FILEDEF, it creates the DMSFLD MODULE. Then, it automatically reinvokes itself to create the DMSFLE MODULE.

Change in Source	Requires Regeneration of Module/Segment	EXEC Procedure To Use
DMSGLB	GLOBAL	CMSGEND
DMSGLO	GLOBALV	CMSGEND
DMSGMF	CMSDOS	DOSGEN
DMSGND	GENDIRT	CMSGEND
DMSGTM	CMSDOS	DOSGEN
DMSGVE	CMSDOS	DOSGEN
DMSHDI	HNDINT	CMSGEND
DMSHDS	HNDSVC	CMSGEND
DMSHLB	DMSHLP	CMSGEND
DMSHLD	DMSHLP	CMSGEND
DMSHLE	DMSHLP	CMSGEND
DMSHLI	DMSHLP	CMSGEND
DMSHLP	DMSHLP	CMSGEND
DMSHLS	DMSHLP	CMSGEND
DMSICP	IOCP	CMSGEND
DMSIDE	IDENTIFY	CMSGEND
DMSIMA	IMAGEMOD	CMSGEND
DMSLAB	CMSDOS	DOSGEN
DMSLBD	LABELDEF	CMSGEND
DMSLBM	MACLIB	CMSGEND
DMSLBT	TXTLIB	CMSGEND
DMSLBR[4]		
DMSLCK	CMSDOS	DOSGEN
DMSLDF	CMSDOS	DOSGEN
DMSLDS	LISTDS	CMSGEND
DMSLIC	CMSDOS	DOSGEN
DMSLKD[4]		
DMSLLU	LISTIO	CMSGEND
DMSMCM	CMSDOS	DOSGEN
DMSMDP	MODMAP	CMSGEND
DMSMVE	MOVEFILE	CMSGEND
DMSMVG[5]		
DMSNXD	NUCXDROP	CMSGEND
DMSNXL	NUCXLOAD	CMSGEND
DMSNXM	NUCXMAP	CMSGEND
DMSOPL	CMSDOS	DOSGEN
DMSOPT	OPTION	CMSGEND

[4] Update procedures are not provided for DMSLBR and DMSLKD. DMSLBR is shipped as part of CMSBAM DOSLIB. DMSLKD is contained in the LKED module.

[5] DMSMVG is loaded into free storage by DMSMVE. The file DMSMVG TEXT must be filemode 2 on the system disk.

Change in Source	Requires Regeneration of Module/Segment	EXEC Procedure To Use
DMSOR1	CMSDOS	DOSGEN
DMSOR2	CMSDOS	DOSGEN
DMSOR3	CMSDOS	DOSGEN
DMSOSR	OSRUN	CMSGEND
DMSOVR	SVCTRACE	CMSGEND
DMSOVS	DMSOVS	CMSGEND
DMSPOL	PROP	CMSGEND
DMSPOP	PROP	CMSGEND
DMSPOQ	PROP	CMSGEND
DMSPOR	PROP	CMSGEND
DMSPOS	PROP	CMSGEND
DMSPRE	PRELOAD	CMSGEND
DMSPRT	PRINT	CMSGEND
DMSPRV	PSERV	CMSGEND
DMSPUN	PUNCH	CMSGEND
DMSRDC	READCARD	CMSGEND
DMSRDR	RDR	CMSGEND
DMSRNE	RENUM	CMSGEND
DMSROS[6]		
DMSRPG	CMSDOS	DOSGEN
DMSRRV	RSERV	CMSGEND
DMSRSF	RXSYSFN	CMSGEND
DMSRSV	RESERVE	CMSGEND
DMSSCR	EDIT[7]	CMSGEND
DMSSET	SET	CMSGEND
DMSSPR	SETPR	CMSGEND
DMSSRT	SORT	CMSGEND
DMSSRV	SSERV	CMSGEND
DMSSSK	SETKEY	CMSGEND
DMSSTX	CMSDOS	DOSGEN
DMSSUB	CMSDOS	DOSGEN
DMSSVL	CMSDOS	DOSGEN

[6] DMSROS is loaded into storage by DMSACM when an OS/DOS disk is accessed. The file DMSROS TEXT must be filemode 2 on the system disk.

[7] When the CMSGEND EXEC procedure is invoked for EDIT, it creates the EDIT module. Then, it automatically reinvokes itself to create the EDMAIN module.

Change in Source	Requires Regeneration of Module/Segment	EXEC Procedure To Use
DMSSYN	SYNONYM	CMSGEND
DMSTMA	TAPEMAC	CMSGEND
DMSTPD	TAPPDS	CMSGEND
DMSTPE	TAPE[8]	CMSGEND
DMSTPF	TAPE[8]	CMSGEND
DMSTPG	TAPE[8]	CMSGEND
DMSTPH	TAPE[8]	CMSGEND
DMSTYP	TYPE	CMSGEND
DMSUPD	UPDATE	CMSGEND
DMSUTL	LOADLIB	CMSGEND
DMSVAN	CMSAMS	VSAMGEN (AMS)
DMSVAS	CMSAMS	VSAMGEN (AMS)
DMSVAX	CMSAMS	VSAMGEN (AMS)
DMSVBM[9]		
DMSVIP	CMSVSAM	VSAMGEN (VSAM)
DMSVIS	CMSDOS	DOSGEN
DMSVLT	CMSDOS	DOSGEN
DMSVVN	CMSVSAM	VSAMGEN (VSAM)
DMSVVS	CMSVSAM	VSAMGEN (VSAM)
DMSXCP	CMSDOS	DOSGEN
DMSXMS	DMSXMS[10]	CMSGEND
DMSXRE	DMSXRE[10]	CMSGEND
DMSZAP	ZAP	CMSGEND
DMSZIT	EDIT[7]	CMSGEND

[7] When the CMSGEND EXEC procedure is invoked for EDIT, it creates the EDIT module. Then, it automatically reinvokes itself to create the EDMAIN module.

[8] When the CMSGEND EXEC procedure is invoked for TAPE, it creates the TAPE module. Then, it automatically reinvokes itself to create the DMSTPF, DMSTPG, and DMSTPH modules.

[9] Update procedures are not provided for DMSVBM. DMSVBM is shipped as part of the CMSBAM DOSLIB.

[10] All EDIT source files, except DMSXMS and DMSXRE, are contained within the CMS nucleus.

Whenever a PTF is applied to a System Product Interpreter source program, the EXECUPDT command is used to create an executable version of the program. This command can only be used with System Product Interpreter programs. The following table indicates the options which should be selected in each case:

Change in EXEC	EXEC Procedure To Use	Indicated Options
ALL $XEDIT	EXECUPDT	fn CTL HIST SID
CANCEL $XEDIT	EXECUPDT	fn CTL HIST SID
CAPPEND $XEDIT	EXECUPDT	fn CTL HIST SID
EXECUPDT $EXEC	EXECUPDT	fn CTL HIST SID
EXECUTE $XEDIT	EXECUPDT	fn CTL HIST SID
PREFIXX $XEDIT	EXECUPDT	fn CTL HIST SID
PRFSHIFT $XEDIT	EXECUPDT	fn CTL HIST SID
PRFSHOW $XEDIT	EXECUPDT	fn CTL HIST SID
PROPHCHK $EXEC	EXECUPDT	fn CTL HIST
PROPLGER $EXEC	EXECUPDT	fn CTL HIST
PROPPCHK $EXEC	EXECUPDT	fn CTL HIST
PROPRTCV $EXEC	EXECUPDT	fn CTL HIST
PROPST $EXEC	EXECUPDT	fn CTL HIST
RGTLEFT $XEDIT	EXECUPDT	fn CTL HIST SID
SPLTJOIN $XEDIT	EXECUPDT	fn CTL HIST SID
STATUS $XEDIT	EXECUPDT	fn CTL HIST SID

"fn" references an UPDATE control file that controls the application of multiple UPDATE files to the source input file. See "Chapter 10. EXEC Procedures and Command Format Summaries" and the*VM/SP CMS Command and Macro Reference* for more information about the EXECUPDT command.

If you must regenerate the CMS nucleus, see "Updating CMS." If you must regenerate the CMSVSAM, CMSAMS, CMSDOS or CMSBAM segments, see "Chapter 5. Loading and Saving Discontiguous Saved Segments." All the other EXEC procedures for generating segments are described in "Chapter 10. EXEC Procedures and Command Format Summaries."

CMS and CMSL should be resaved whenever the system S-disk is updated. This will ensure that the S-disk directory, which is a part of the saved CMS nucleus, correctly matches the physical S-disk directory. See "Updating CMS."

See "Creating CMS Disk-Resident Modules," if a CMS module must be regenerated. A description of CMSGEND EXEC is presented in "CMSGEND" for further information.

If you apply a PTF to certain CP source programs, the corresponding CMS modules must also be regenerated to run properly. The source name, module name, and procedures used for regenerating the modules are shown in the following table[23].

Change in Source	Requires Regeneration of Module	EXEC Procedure(s) To Use
DMKDDR	DDR	GENERATE, CMSGEND
DMKDIR	DIRECT	GENERATE, CMSGEND
DMKFMT	None, no CMS execution	GENERATE
DMKNMT	IMAGELIB	CMSGEND
DMKIMG	GENIMAGE	CMSGEND
DMKRND	NCPDUMP	CMSGEND
DMSARN	ASM3705	INSTEP
DMSARD	ASM3705	INSTEP
DMSARX	ASM3705	INSTEP
DMSGRN	GEN3705	CMSGEND
DMSNCP	SAVENCP	CMSGEND
VMFDATE	VMFDATE	CMSGEND
VMFDOS	VMFDOS	CMSGEND
VMFLOAD	VMFLOAD	CMSGEND
VRSIZE	VRSIZE	CMSGEND
DMKTRR	TRAPRED	CMSGEND

See Appendix D, "Generating and Loading the 3704/3705 Control Program" for a description of the INSTEP EXEC procedure. All other EXEC procedures are described in "Chapter 10. EXEC Procedures and Command Format Summaries"if you need to examine them.

[23] Use this appendix only when applying updates and local fixes to your system. When corrective service is shipped via PUT, all modules are regenerated and replaced on a one-to-one basis.

Appendix B. A Sample EXEC Procedure for Copying VSE Macros into a CMS MACLIB

You may wish to create the following EXEC procedure, DOSMAC, which will aid you in creating a VSE macro library under CMS.

Note: This procedure has not been formally tested by IBM; it is presented here for your convenience only.

To execute the following EXEC procedure, you must be in CMS/DOS mode. If a private source statement library is to be used, the appropriate ACCESS, ASSGN, and DLBL commands must be issued, specifying the VSE disk where that library resides. The procedure creates a DSERV listing on your CMS disk and uses the source statement directory listing to create an EXEC file that issues a separate ESERV command for each VSE macro. You can then use the CMS Editor or System Product Editor to delete all the ESERV commands for macros you do not wish to move at this time. The procedure then creates a CMS macro library with a MACLIB filename specified by you. If you do not specify a filename, the default is DOSMAC.

Note: If you have too many VSE macros to move to your CMS disk, the MACLIB build process may exceed the CMS file system limitation and abnormally terminate. All macros prior to the one that caused the error message probably were cataloged correctly. Reinvoke the EXEC procedure and then use the CMS Editor or System Product Editor to delete the ESERV commands for all the macros previously cataloged. You must also specify some other filename (such as DOSMAC2) for this new macro library.

Alternatively, if you want to avoid the abnormal termination of the MACLIB build process, you may want to delete some or all of the ESERV commands for the following VSE macros the first time you invoke this EXEC procedure:

```
BTMOD
CDMOD
DAMOD
DAMODV
FOPT
IOINTER
IOTAB
ISMOD
MCRAS
MTMOD
SGCCWT
SGEND
SGPMAIN
SGPSUB
SGSVC
COBBG
COBF2
```

Note: Check a DSERV listing and delete the ESERV commands for the largest VSE macros first. Then manually create a second set of ESERV commands, specifying those macros not included in the first CMS MACLIB.

Creating the DOSMAC EXEC Procedure

Issue the following command to create a EXEC procedure called DOSMAC EXEC:

```
xedit dosmac exec
```

Enter the INPUT subcommand to get into input mode and key in the following lines.

```
    &CONTROL OFF
    &GENSWT = 0
    CP PURGE RDR ALL
    CP SP 9 * CLASS A
    &TYPE ENTER THE ADDRESS OF YOUR SYSRES VOLUME ( DEFAULT = 350 )
    &READ ARGS
    &IF &INDEX EQ 0 ACCESS 350 Z
    &IF &INDEX NE 0 ACCESS &1 Z
    SET DOS ON Z  ( VSAM
    &TYPE IF YOU WISH TO ASSGN AND DLBL A PRIVATE SOURCE STATEMNT LIBRARY
    &TYPE NOW IS THE TIME ( ENTER YOUR ASSGN ). IF YOU DO NOT ENTER A NULL LINE
    &READ
    &TYPE A DLBL IS ALSO REQUIRED FOR SSL
    &READ
    -MACGEN &CONTINUE
    &TYPE ENTER THE NAME OF THE MACLIB TO BE CREATED THE DEFAULT IS DOSMAC
    &READ ARGS
    &IF &INDEX EQ 0 &LIB = DOSMAC
    &IF &INDEX NE 0 &LIB = &1
*
*   The output of the DSERV command is spooled to
*   your virtual card reader and is read in as $ESER EXEC.
*
    CP SPOOL CONS START NOTERM
    DSERV SD ( TERM
    CP SPOOL CONS STOP TERM
    CP CLOSE 9
    READ $ESER EXEC
*
*   The $ESER EXEC file is copied, edited, and formatted
*   is a CMS EXEC file.  All DSERV header and trailer lines
*   are deleted.
*
    COPYFILE $ESER EXEC A $ESERV EXEC A ( LRECL 80  REPLACE
    &BEGSTACK
    DEL 9
    F CP
    DEL 5
    TOP
    C / /&1 &2/*
    FILE
    &END
    EDIT $ESERV EXEC
    ERASE $ESER EXEC
    -STACKER &CONTINUE
    &BEGTYPE
```

```
*
*   If you wish to delete any of the generated ESERV commands, enter
*   any nonblank character.  If you do not wish to delete any ESERV
*   commands (or after you have deleted them), enter a null line.
*
*   If you wish to alter the list of macros, now is the time to do so.
*   You may bypass alteration by entering a null line or else enter a
*   non-blank character to begin alteration.  Alteration is accomplished
*   via editing the exec file containing the macro names.  You must
*   issue the edit subcommand "file" to re-enter this exec and continue.
*
    &END
    &READ ARGS
    &IF &INDEX NE 0 EDIT $ESERV EXEC
    &CONTROL ALL
*
*   Stack the remaining lines of the $ESERV EXEC
*   in the console stack.
*
    EXEC $ESERV &STACK SPACE
    ASSGN SYSIN A
    ASSGN SYSLST PRINTER
    ASSGN SYSPCH PUNCH
    CP SPOOL D TO *
    &CONTROL ALL
    -GETNEXT &CONTINUE
    &READ ARGS
*
*   Read a line from the console stack and check that
*   the first letter begins with E (for ESERV).
*   If not an E, ignore the line and read the next one.
*
    &IF &2 NE E &GOTO -STAKTST
*
*   If it is an E, create a DSPCH fn for this macro.
*   Note:  PUNCH or DSPLY may be substituted for DSPCH.
*
    &STACK LIFO FILE
    &STACK LIFO C /$/ / 4
    &STACK LIFO TOP
    &STACK LIFO I $DSPCH &3
    EDIT &3 ESERV
*
*   Execute the ESERV command.  The re-edited macro is
*   spooled to your virtual card reader.
*
    EXEC ESERV &3
    ERASE &3 ESERV
```

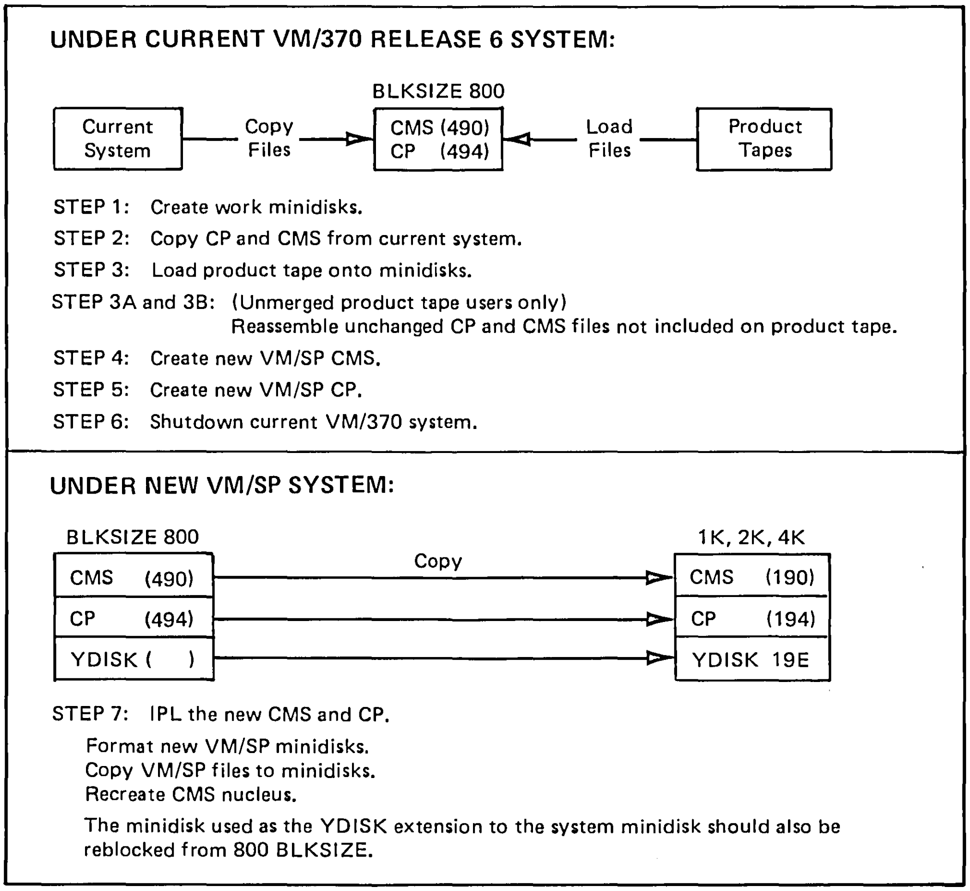
```
*
*   Read the macro file onto the CMS disk.
*   Delete the CATALS statement.
*
    READ &3 MACRO
    &STACK LIFO FILE
    &STACK LIFO DEL
    &STACK LIFO BO
    &STACK LIFO DEL
    &STACK LIFO L /CATALS/
    EDIT &3 MACRO
    &IF &GENSWT NE 0 &GOTO -MACADD
    &GENSWT = 1
    MACLIB GEN &LIB &3
    ERASE &3 MACRO
    &GOTO -STAKTST
    -MACADD &CONTINUE
*
*   Add the macro to the indicated CMS macro library.
*
    MACLIB ADD &LIB &3
    ERASE &3 MACRO
    &IF &READFLAG EQ STACK &GOTO -GETNEXT
    -FINALE &CONTINUE
    &STACK QUIT
    &BEGTYPE
     THE MACLIB &LIB HAS BEEN CREATED AND THE FOLLOWING IS A MAP OF THE LIBRARY
    &END
    &STACK MACLIB MAP &LIB ( TERM
    &EXIT
    -STAKTST &CONTINUE
    &IF &READFLAG EQ STACK &GOTO -GETNEXT
    &GOTO -FINALE
```

The ESERV process may take a substantial length of time, up to several hours for a large macro library. A detailed description of the ESERV command can be found in the *VM/SP CMS Command and Macro Reference.* For more information on how to use the ESERV command, see "Appendix D: Sample Terminal Session for VSE Programs" in the *VM/SP CMS User's Guide.* Refer to the *Guide to the DOS/VSE Assembler*, GC33-4024 for a detailed description of the VSE ESERV control statements.

Appendix C. VM/SP Without the Starter System and the Merged Product Tapes

Generating a VM/SP System from a VM/370 Release 6 System

You can use the following steps to generate a VM/SP system from a VM/370 Release 6 system. At the end of these steps you will have a VM/SP system without current service. You can then apply VM/SP service as described in the PUT Document and *Memo to Users* supplied with the service tape. The figure below shows the steps involved in installing VM/SP on a Release 6 SCP system.



STEP 1: Create work minidisks.

STEP 2: Copy CP and CMS from current system.

STEP 3: Load product tape onto minidisks.

STEP 3A and 3B: (Unmerged product tape users only)
Reassemble unchanged CP and CMS files not included on product tape.

STEP 4: Create new VM/SP CMS.

STEP 5: Create new VM/SP CP.

STEP 6: Shutdown current VM/370 system.

STEP 7: IPL the new CMS and CP.

Format new VM/SP minidisks.
Copy VM/SP files to minidisks.
Recreate CMS nucleus.

The minidisk used as the YDISK extension to the system minidisk should also be reblocked from 800 BLKSIZE.

Figure 22. VM/SP Installation on a Release 6 SCP System

Step 1. Create and Format the New CMS System Disk and CP Staging Area

Assuming minidisk addresses of 490 and 494 for the new CMS system disk and CP staging area, respectively, the following sample sizes may be used for reference.

	3330 cyl.	**3340 cyl.**	**3350 cyl.**	**3375 cyl.**	**3380 cyl.**	**FB-512 blocks**
CMS SYSTEM DISK (490)	95	223	45	70	36	43320
CP STAGING AREA (494)	45	110	22	32	18	20520

Remember to RECOMP the CMS minidisk to reserve space for the CMS nucleus.

Step 2. Copy CP and CMS Files from Your Current System

Ensure that MAINT's 190 and 194 are at the VM/370 Release 6, PLC 13 level. Refer to the *Program Directory* distributed with the product tape for special considerations.

When you are at the correct level, copy the 190 and 194 minidisks to the 490 and 494 minidisks, respectively. If you have an IBM processor and the Merged Product Tapes, you need copy only the following modules (if desired):

RSCS modules (DMTxxx) from 194 to 494 minidisk

IPCS modules (DMMxxx) and the following modules from 190 to 490 minidisk:

```
DUMPSCAN
PROB
STAT
SUMMARY
VMFDUMP
VMFDUMP2
```

Step 3. Load the Product Tape

You should load the Product Tape to get the current VM/SP text files onto the 490 and 494 minidisks. Refer to "Chapter 1. Introduction to VM/SP Installation" for the product tape format and other installation ideas.

Step 3A. (VM/SP Only Product Tape) Reassemble the CP Release 6 Modules

If you have a VM/SP Only Product Tape, you should reassemble the following CP modules from Release 6. Before issuing the command, be sure the TEXT AUXR60 entry is included in your CNTRL file.

Issue:

```
vmfasm xxxxxxdmksp□
```

where:

xxxxxx is:

```
DMKALG   DMKISM   DMKSIX   DMKVCA   DMKEIG   DMKNMT   DMKUCC
DMKERM   DMKRND   DMKUCB   DMKIMG   DMKSEV   DMKUCS
```

and □ is: null (UP), 'A' (AP), or 'M' (MP) as appropriate.

Step 3B. (VM/SP Only Product Tape) Reassemble the CMS modules

CMS modules using: If you have a VM/SP Only Product Tape, you should reassemble the following CMS modules using:

```
vmfasm xxxxxx dmssp
```

where:

xxxxxx is:

```
DMSARD   DMSASD   DMSCWT   DMSEDC   DMSEDF   DMSGIO   DMSHDI
DMSHDS   DMSINM   DMSIOW   DMSLOA   DMSLSY   DMSMDP   DMSOR1
DMSOR2   DMSOR3   DMSSSK   DMSTQQ   DMSZAT   DMSZNR   DMSZUS
```

Step 4. Create the New VM/SP CMS nucleus

To create the new VM/SP CMS nucleus, issue:

```
access 190 a
access 490 e
```

The VM/SP modules on 490 should *not* be used at this time. Issue:

```
rename punch module e2 pnch = =
rename rename module e2 renm = =
spool pun * nohold
access 490 a
vmfload cmsload dmssp
```

Note: Do not issue any CMS commands while 490 is accessed as A (a wrong or incomplete module may be loaded).

IPL the reader (00C) when the load deck is complete and load the new CMS nucleus onto minidisk 490. Rename the modules that were renamed previously. Issue:

```
access 190 a
access 490 e
rename pnch module e2 punch = =
rename renm module e2 rename = =
```

Step 5. Create the New VM/SP CP Nucleus

Def STOR 3M

With most system levels, an IPL of the newly created CMS followed by an access of the CP text disk should be sufficient to assemble and build your new CP system.

```
ipl 490
access 494 e
```

If this does not work on your system, do the following:

```
ipl 190
access 490 b/a * maclib
access 490 c/a * loader
access 494 e
```

Create new DMKSYS, DMKSNT, and DMKRIO text files using the appropriate CP CNTRL file on 494.

```
vmfasm DMKxxx dmksp□
```

Create a new CP nucleus:

```
spool pun * nohold
vmfload nnload dmksp□
```

where:

nn is 'CP', 'AP', 'VR', or 'AV' as appropriate.

□ is null (UP), 'A' (AP), or 'M' (MP) as appropriate.

```
ipl 00C
```

After the nucleus is loaded, close the printer and read-in (save) the NUCMAP.

Step 6. Shutdown the System

Step 7. IPL the Newly Created CP and CMS Nuclei

At this time you should IPL the new CP nucleus and the CMS nucleus created in Steps 1 through 6 and create minidisks formatted under the new CMS system.

IPL the new CP nucleus from your processor according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)

Copy the VM/SP files from the 490 and 494 minidisks to new minidisks.
Re-create the CMS nucleus for the new CMS minidisk.

```
IPL the new 190
```

Recreate the ASSEMBLE module for the new CMS. Issue:

```
access 190 a

cmsgend assemble
```

Enter:

```
'eob' (carriage return)
```

when asked about the target of the GENDIRT command. The ASSEMBLER auxiliary directory will use the 'S' disk.

You now have an operational VM/SP system without current service. Include the VM/SP service as described in the PUT Document and *Memo to Users.*

Appendix D. Generating and Loading the 3704/3705 Control Program

Several commands and EXEC procedures generate and load the 3704/3705 control program. These commands and EXEC procedures are executed in a CMS virtual machine. The commands are part of the VM/SP system and are distributed with it.

A special version of the IBM 3704/3705 Emulation Program Support Package for OS/VS, Order No. 5744-AN1 EP/VS SCP is available from PID for use under VM/SP. This version of the 3704/3705 package contains two CMS EXEC procedures for generating and loading the 3704/3705 control programs.

This chapter describes the step-by-step procedure for generating and loading the 3704/3705 control program. Each EXEC procedure and command is described as it is used. The action required at each step is summarized first and then explained in detail. "Part 1. Planning for System Generation" of the *VM/SP Planning Guide and Reference* lists all the documentation, physical devices, programming, and other materials you need before starting to generate the 3704/3705 control program.

Step 1. Log on the VM/SP System

VM/SP supports the EP type of control program. The VM/SP system that you load also must have been generated with:

- The IBM 3704 or 3705 Communications Controllers specified on a RDEVICE system generation macro.
- The NAMENCP macro coded to create an entry in the VM/SP system name table (DMKSNT) for the 3704/3705 control program.
- Space reserved on a CP-owned volume to contain a copy of the 3704/3705 control program.

These VM/SP system generation requirements are described in "Part 1. Planning for System Generation" of the *VM/SP Planning Guide and Reference.*

Step 2. Set Up a CMS Virtual Machine

You must IPL CMS in a virtual machine and be sure that the necessary devices are attached.

The 3704/3705 control program is generated using commands and EXEC procedures that execute in a CMS virtual machine. The CMS virtual machine must have the following resources:

- At least 1024K of virtual storage.[24]
- One tape drive (9 track, 800 or 1600 bpi).
- Space available on the CMS A-disk (120 cylinders of a 3330 disk, all 203 cylinders of a 2314 disk, 300 cylinders of a 3340 disk, 60 cylinders of a 3350 or 3380 disk, 80 cylinders of a 3375 disk, or 7200 pages of an FB-512 disk.

If the CMS virtual machine does not have these resources, use the CP DEFINE command to redefine the size of your virtual storage or send a message to the operator requesting the needed tape or disk device be attached.

Be sure that there are no files on the A-disk with a filetype of COPY or TEXT. Use the CMS RENAME command to temporarily change such filetypes. A naming conflict can terminate the installation procedure for the distribution tape.

You need CP command privilege classes A, B, or C to install the 3704/3705 control program. Check with the system administrator to ensure that your VM/SP directory entry has the appropriate command privileges.

Step 3. Load the IBM 3704/3705 Control Program Distribution Tape Files onto a CMS Disk that has been accessed as filemode A.

Note: Do **not** access the CMS System-Disk (190) as filemode A.

Use CMS commands to position the distribution tape at the proper file and to create CMS disk files from the tape files. The first file created from the tape files is an EXEC procedure that processes the rest of the tape files and creates the CMS disk files.

Have the distribution tape mounted and attached to your virtual machine. The distribution tape contains ten physical files, separated by tape marks. The tenth file contains the INSTEP and ARNGEND EXEC procedures, which create the necessary CMS files from the other tape files.

Use the CMS TAPE command to position the tape at the beginning of the tenth file:

```
tape rew
tape fsf 9
```

[24] Depending on the size and options of the EP Program being built, and depending on the blocksize and track size of DASD used, the minimum amount of virtual storage might have to be increased. Verify that the 3705 assembler has been built; then use a larger virtual storage size, e.g., 4096K or greater to continue.

Then, use the CMS TAPPDS command to create the INSTEP EXEC A1 and ARNGEND EXEC A1 files from the tenth file:

```
tappds * exec
```

If the files are successfully created, the responses

```
FILE 'ARNGEND EXEC A1' COPIED
FILE 'DMSARD  EXEC A1' COPIED[25]
FILE 'DMSARX  EXEC A1' COPIED[25]
FILE 'DMSGRN  EXEC A1' COPIED[25]
FILE 'DMSTMA  EXEC A1' COPIED[25]
FILE 'INSTEP  EXEC A1' COPIED
```

appear on your terminal. Before invoking the INSTEP EXEC procedure do the following:

1. XEDIT INSTEP EXEC, deleting the RENAME commands for DMSARD, DMSARX, DMSGRN, and DMSTMA.

2. Erase the DMSARN, DMSARX, DMSGRN, and DMSTMA EXEC files.

3. Access the CMS System-Disk (190) as T/A to get access to the mode 1 files.

Invoke the INSTEP EXEC procedure to load all the necessary files and generate the 3705 Assembler:

```
instep
```

The INSTEP EXEC procedure generates the 3705 Assembler and creates the macro and text libraries that are needed to generate a 3704/3705 control program. The INSTEP EXEC procedure sends messages to the terminal to indicate its progress.

INSTEP issues the message:

```
BUILD STAGE ONE MACLIB
LOADING 'GEN3705 MACLIB'
```

and uses the third tape file to create the CMS file GEN3705 MACLIB A1. INSTEP also issues the messages:

```
BUILD STAGE TWO MACLIBS
LOADING 'MAC3705 MACLIB'
```

using the fifth tape file to create the CMS file MAC3705 MACLIB A1. Using the sixth tape file, INSTEP creates the CMS file OBJ3705 MACLIB A1, and issues the messages:

```
BUILD STAGE TWO TXTLIB

LOADING 'OBJ3705 MACLIB'
RENAME OBJ3705 MACLIB A1 OBJ3705 TXTLIB A1
```

[25] These files may not be present on later releases of the EP/VS distribution tape.

Finally, INSTEP issues the message:

```
LOAD 3705 ASSEMBLR FILES
```

and loads the assembler text files from tape via the TAPPDS command. The files copied are listed off in messages in the form:

```
FILE 'fn EPTAPE A1' COPIED
```

The ARNGEND EXEC procedure is invoked by INSTEP to generate the 3705 Assembler, after issuing the message:

```
BUILD 3705 ASSEMBLR MODULES.
```

The ARNGEND EXEC procedure displays the following status and error messages:

```
ENTER TARGET DISK6MODE FOR 3705 ASSEMBLR MODULES
DEFAULTS TO S-DISK IF NONE ENTERED
```

You enter the mode letter of the disk that will contain the 3705 assembler modules when the assembler is used. This may be a different disk then the one on which the modules now reside. If you enter a mode letter, ARNGEND uses that mode letter as the "targetmode" operand of the GENDIRT command when it creates the auxiliary directory for the 3705 assembler. If you do not specify a mode letter, S is assumed by the GENDIRT command.

If the 3705 assembler text files are not loaded successfully, or if the assembler generation procedure fails, the following message appears:

```
ASM3705 GEND FAILED
```

When the last message:

```
END OF EPTAPE INSTALL
```

appears on the terminal, the distribution tape is no longer needed. At this time, the 3705 Assembler program, the macro libraries for the Stage 1 and Stage 2 generation procedures, and the text library for the Stage 2 generation procedure all exist on the CMS A-disk.

Note: You may find it helpful to dump the contents of the A-disk to tape at this time. If you save the tape dump, you have the pre-Stage 1 files. If errors are later encountered, you may need these files.

Step 4. Code the 3704/3705 Control Program Macro Instructions

Code the 3704/3705 control program macro instructions and place them in a CMS file. Use the System Product Editor to create the file, which must have a filetype of ASM3705. VM/SP recommends that you assign the same filename to this CMS file as was specified previously in the NAMENCP macro. If the SAVE option is to be specified on the GEN3705 command, the filename must be the same. This ASM3705 file is used as input to Stage 1 of the 3704/3705 control program generation procedure.

Use the *3704 and 3705 Generation and Utilities Guide* to code the macro instructions. Follow the macro instruction formats described in that publication except where suggestions and requirements are indicated in the following paragraphs.

BUILD Macro Instruction: The BUILD macro must be the first macro in the CMS file. Figure 23 lists the operands which VM/SP requires, recommends, or does not support. For all other operands, refer to the *3704 and 3705 Generation and Utilities Guide.*

Operand	Comments

LOADLIB=dsname
OBJLIB=dsname
	Required by the BUILD macro, but does not apply to VM/SP. Specify a valid dsname.

JOBCARD= $\left\{\begin{array}{l}\textit{YES}\\ \text{NO}\end{array}\right\}$	VM/SP recommends JOBCARD=YES for EP.
NEWNAME= $\left\{\begin{array}{l}\text{NCP001}\\ \text{PEP001}\\ \text{VMEP01}\\ \text{symbol}\end{array}\right\}$	VM/SP requires that the value of NEWNAME be the same as the name previously specified in the NAMENCP macro and the name that subsequently will be specified in the SAVENCP command. Also, if the GEN3705 command is to be issued with the SAVE option, the value of NEWNAME must be the same as the "fname" specified on the GEN3705 command.
QUALIFY= $\left\{\begin{array}{l}\text{symbol}\\ \text{NONE}\\ \textit{SYS1}\end{array}\right\}$	VM/SP requires the default value.

UT1=dsname
UT2=dsname
UT3=dsname
	VM/SP ignores these operands.

Figure 23. BUILD Macro Operands for VM/SP

CSB Macro Instruction: The CSB macro instruction is required. See the *3704 and 3705 Generation and Utilities Guide* for more information about coding the CSB macro instruction.

GROUP and LINE Macro Instructions: These macros describe the physical and logical configuration of the communications network accessed through the 3704/3705 control program. Since VM/SP does not support either multi-drop lines or cluster control units, the 3704/3705 configuration for VM/SP is generally simple, with only one GROUP macro for each communications scanner. For VM/SP, it is often easiest to specify most of the operands of the LINE macro on the GROUP macro. The *3704 and 3705 Generation and Utilities Guide* describes the GROUP and LINE macro instructions in detail and lists all the operands of the configuration macros, telling you where each operand is described and also where it may be coded.

VM/SP requires the DUPLEX and FEATURE operands. These operands allow VM/SP to detect and respond to a terminal attention interrupt and to recognize when a data set has been hung up. For the GROUP macro, VM/SP requires the default value for the REPLYTO operands and recommends the default value for the TEXTTO operand.

GENEND Macro Instruction: The GENEND macro indicates the end of the 3704/3705 macro input file. It must be the last macro in the CMS file you are building as input to Stage 1.

Special Macro Coding Considerations for the Emulation Program (EP): There are no strict dependencies between the host access method and the emulation program; consequently, few guidelines are necessary for an emulation program generation. However, be careful when configuring emulator lines for CPT-TWX terminals. While VM/SP normally accepts incoming calls from either 1050 or 2741 terminals on the same physical line, that same line cannot be used for CPT-TWX terminals. When generating the VM/SP system, ensure that the hardware configuration specified in the CP module DMKRIO matches the configuration of the Emulation Program for CPT-TWX lines; the exact configuration of 1050 and 2741 lines is not critical.

Note: The base address of the 3704/3705 (the address used to load and/or dump the control program) can never be specified for use as a telecommunications line. VM/SP treats the base address as a separate entity for use only during the load and dump operation.

Step 5. Define the Macro and Text Libraries

The macro and text libraries created from the distribution tape in Step 3 must be made available to CMS. One macro library (GEN3705) is needed for the Stage 1 generation procedure and one macro library (MAC3705) and one text library (OBJ3705) are needed for the Stage 2 generation procedure. It is easiest to define all the libraries before starting Stage 1. Use the CMS GLOBAL command:

```
global maclib gen3705 mac3705
global txtlib obj3705
```

Step 6. The Stage 1 Generation Procedure

The Stage 1 generation procedure accepts the CMS file you created in Step 4 as input and produces the Stage 2 input file that is needed in Step 7.

The Stage 1 generation procedure is performed by invoking the 3705 Assembler to process the 3704/3705 control program macro instructions. It produces one file with the same filename as the input file and with a filetype of TEXT. This TEXT file contains 3705 Assembler source statements and job control language (JCL) statements.

The ASM3705 Command: Use the CMS ASM3705 command to invoke the 3705 Assembler to assemble the macro instruction file. The 3705 Assembler processing and output are controlled by the options selected.

The format of the ASM3705 command is:

```
ASM3705     fn [(options... )]
            Options:
            [XREF (FULL ) ]  [RENT  ]  [DECK  ]  [LOAD  ]
            [XREF (SHORT) ]  [NORENT]  [NODECK]  [NOLOAD]
            [NOXREF       ]

            [LIST  ]  [LINECOUN (55)]  [PRINT  ]
            [NOLIST]  [LINECOUN (nn)]  [DISK   ]
                                       [NOPRINT]
```

where:

fn specifies the filename of the source file to be assembled. This source file contains the 3704/3705 control program macro instructions. The file must have a filetype of ASM3705 and fixed-length, 80-character records.

Options:

If duplicate or conflicting options are specified, the last one entered in the command line is the one in effect.

`XREF (FULL)` includes in the assembler listing a cross-reference table of all symbols used in the assembly. This includes symbols that are defined but never referenced. The assembler listing also contains a cross-reference table of literals used in the assembly.

`XREF (SHORT)` includes in the assembler listing a cross-reference table of all symbols used in the assembly. Any symbols defined but not referenced are not included in the table. The assembler listing contains a cross-reference table of literals used in the assembly.

`NOXREF` does not print the cross-reference table.

`RENT` checks the source file to see if it satisfies reentrancy requirements.

`NORENT` suppresses the check for satisfaction of reentrancy requirements.

`DECK` spools the output object module, fn TEXT, to the punch.

`NODECK` suppresses the spooling of the output object module, fn TEXT, to the punch.

`LOAD` creates a TEXT file on disk for the program that was assembled.

`NOLOAD` suppresses the creation of a TEXT file on disk for the program that was assembled.

`LIST` produces a LISTING file.

`NOLIST` produces no LISTING file.

`PRINT` spools the LISTING file to the printer.

`DISK` puts the LISTING file on disk.

`NOPRINT` produces no LISTING file.

`LINECOUN nn` specifies the number of lines per output printer page. A default of 55 lines is assumed.

Note: All of the options of the 3705 XF Assembler are supported and may be used with the ASM3705 command, with the exception of ALIGN/NOALIGN and TEST/NOTEST.

Files Created by the ASM3705 Command

TEMPORARY WORKFILES: Three files are temporarily created for each assembly.

```
fn SYSUT1
fn SYSUT2
fn SYSUT3
```

Any existing files with the same file identifiers are erased at the beginning of the assembly. These files are placed on the read/write disk with the most available space. Work space is automatically allocated as needed during the assembly and returned to available status when the assembly is complete. Insufficient space causes abnormal termination of the assembly.

PERMANENT FILES: One or two permanent files may be created during a successful assembly.

```
fn TEXT
fn LISTING
```

The fn TEXT file contains the output object module if the LOAD option is in effect. The fn LISTING file contains a listing of source statements, assembled machine code, and other associated information based on the options selected. This file is created unless the NOPRINT or NOLIST options are selected. The LISTING and TEXT files are placed on (1) the disk from which the source file was read, (2) its parent or (3) the primary disk, unless you created a file definition for these files placing them on a non-DASD. Failure to obtain sufficient space for these files results in abnormal termination of the assembly.

Special Considerations for the Stage 1 Assembly: The Stage 1 assembly can be very lengthy. The amount of time the Assembler takes depends upon the macro options selected and the number of users on the VM/SP system.

The LISTING file produced by the Stage 1 assembly is quite large. If you let the ASM3705 command option default to DISK, much of the space on your A-disk is used. Therefore, VM/SP recommends that you specify the PRINT option when you issue the ASM3705 command. Also, there are many macro expansions that make the LISTING file larger. VM/SP recommends that you insert a 'PRINT NOGEN' assembler statement in front of the first macro instruction in the input file to suppress the printing of the macro expansions and reduce the size of the LISTING file.

You should examine the output of the Stage 1 assembly carefully and produce a list of resource IDs, with their characteristics, for the operations personnel. The cross-reference list for operations should include:

- Resource ID
- Type of resource (line or terminal)
- Type of line (EP-mode or variable)
- Location

Step 7. The Stage 2 Generation Procedure

During the Stage 2 generation procedure the TEXT file produced in Step 6 is scanned. That TEXT file contains several job steps of 3705 Assembler source statements with embedded OS JCL statements. The JCL statements are removed and a unique CMS 3705 Assembler source file is created for each job step in the input file. An EXEC procedure is also created to assemble and link edit the source files. When the EXEC procedure is invoked, it produces the load module file (and, optionally, saves a copy of the control program in page-format on a CP-owned volume).

***The GEN3705 Command*:** Use the CMS GEN3705 command to invoke the Stage 2 service program. Command options let you determine whether or not GEN3705 includes a command in the EXEC procedure to save a copy of the load module on disk, or if GEN3705 invokes the EXEC procedure automatically.

The format of the GEN3705 command is:

```
GEN3705        fname ftype [fmode] [(options... )]

               Options:

                 [RUN  ]  [SAVE  ]
                 [NORUN]  [NOSAVE]
```

where:

`fname` specifies the filename of the Stage 2 input stream produced by the Stage 1 assembly. The file must contain fixed-length, 80-character records.

`ftype` specifies the filetype of the Stage 2 input stream. The filetype is normally TEXT.

`fmode` specifies the filemode.

Options:

If duplicate or conflicting options are specified, the last option entered on the command line is in effect.

`RUN` causes the output EXEC file to be executed at the conclusion of the GEN3705 processing.

`NORUN` suppresses the execution of the output EXEC file.

`SAVE` includes a SAVENCP command in the output EXEC file to create a page-format copy of the 3704/3705 control program on a VM/SP CP-owned volume.

If you are generating a 3705 Emulator control program with a Type 4 channel adapter, do not use the SAVE option; an error message will result from the SAVENCP command. In this case, you must specify the SAVENCP command yourself, specifying the CAMOD option.

NOSAVE does not include the SAVENCP command in the output EXEC file.

Files Created by the GEN3705 Command

Three types of permanent files are created when the GEN3705 command successfully executes: ASM3705, TEXT, and EXEC files.

```
fname00  ASM3705        fnameL0  TEXT          fname EXEC
fname01  ASM3705        fnameL0  TEXT
   .        .              .        .
   .        .              .        .
   .        .              .        .
fnamenn  ASM3705        fnameLn  TEXT
```

A separate ASM3705 file is created for each assembly job step in the Stage 2 input file. Each ASM3705 file created by GEN3705 is given a unique filename of the form 'fnamenn'. The first six characters of the input filename are concatenated with a two-digit number. For example, if the input file is NCP320 TEXT, the output files are NCP32000 ASM3705, NCP32001 ASM3705, ..., NCP320nn ASM3705. These files are used as input to the 3705 Assembler when it is invoked by the Stage 2 EXEC procedure.

The GEN3705 program creates several TEXT files. These files contain only linkage editor control statements, those statements necessary to build the load module file for the 3704/3705 control program. Each of the TEXT files created is given a unique filename of the form 'fnameLn'. The first six characters of the input filename are concatenated with the letter L and a one-digit number. For example, if the input file is NCP320 TEXT, the linkage editor output files are NCP320L0 TEXT, NCP320L1 TEXT, ..., NCP320Ln TEXT.

The filenames assigned to the linkage editor and assembler files must be different. If the filenames are the same, when the ASM3705 files are later assembled, TEXT files will be produced that will have file identifiers that conflict with the linkage editor files.

The EXEC macro file created contains the CMS commands necessary to invoke the ASM3705 command for each of the ASM3705 files, and to subsequently invoke the linkage editor for each of the Assembler TEXT files. If the SAVE option is specified, the EXEC file also contains the SAVENCP command which loads the 3704/3705 control program image into virtual storage and creates the page-format copy of it on a CP-owned volume. The filename of the Stage 2 input file is used as the 'ncpname' operand for the SAVENCP command.

Special Considerations for the Stage 2 Generation Procedure: VM/SP recommends that you specify the RUN option. When the RUN option is specified, GEN3705 stacks a CMS command line to cause the EXEC file to execute following the completion of the GEN3705 program. This technique minimizes the virtual storage overhead during the EXEC file execution.

If you do not specify the SAVE option, you have to explicitly issue the SAVENCP command. If you do specify the SAVE option, be sure that the input file has the same filename as the entry reserved in the system name table. The system name table is created when a NAMENCP macro is issued during a VM/SP system generation. The NAMENCP macro and the building of the system name table are described in "Part 2. Defining Your VM/SP System" of the *VM/SP Planning Guide and Reference.*

Step 8. Invoke the EXEC Procedure Created in Step 7

If you specified RUN on the GEN3705 command, this step is executed for you. If you did not specify RUN on the GEN3705 command, you must invoke the EXEC procedure that the GEN3705 program created.

The EXEC procedure is given the same filename as the GEN3705 input file. It is invoked by entering that filename at the terminal. For example, if the input file is NCP320 TEXT, the EXEC file is named NCP320 EXEC, and can be invoked by issuing:

```
NCP320
```

at the terminal.

This EXEC procedure contains CMS commands that:

- Assemble the 3705 source files (ASM3705 commands).
- Build the TXTLIB that the 3705 Assembler needs (TXTLIB commands).
- Define all the necessary files; such as, the SYSLIB and SYSLMOD files, load libraries, and text libraries (FILEDEF commands).
- Link edit the 3705 text files creating a load module (LKED commands).

You need not issue the ASM3705 and LKED commands that create the 3704/3705 control program load module; the EXEC procedure does that for you. The ASM3705 command is described in Step 6. The FILEDEF, LKED, and TXTLIB commands are described in the *VM/SP CMS Command and Macro Reference.*

Step 9. Save the 3704/3705 Control Program Image on Disk

If you specified SAVE on the GEN3705 command, this step is executed for you. If you did not specify SAVE on the GEN3705 command, you must issue the SAVENCP command yourself.

Note: The VM/SP command privilege class A, B, or C is required to use the SAVENCP command.

***The SAVENCP Command*:** Use the CMS SAVENCP command to read a 3704/3705 control program load module created by the LKED command, and to load it into virtual storage in the CMS user area. Once the load is performed, SAVENCP scans the control program image and extracts the control information required by CP. The control information is accumulated in one or more 4096-byte pages in the CMS user area. When all of the necessary control information is extracted, SAVENCP builds the Communications Controllers Parameter List (CCPARM) and issues the DIAGNOSE X'50' instruction to create the page-format copy of the control program on a CP-owned volume.

The format of the SAVENCP command is:

```
SAVENCP        fname [(options.. )]

               Options:

                  [ENTRY symbol]  [NAME ncpname]
                  [CXFINIT     ]  [fname       ]

                  [LIBE libraryname]  [CAMOD {0}]
                  [fname           ]  [      {1}]
```

where:

`fname` is the filename of the LOADLIB file where the 3704/3705 control program load module resides; unless LIBE is specified, in which case, it specifies the member name of the image within the LOADLIB. This name is used as the ncpname for the DIAGNOSE instruction, unless the NAME option is also specified.

Options:

`ENTRY symbol`
is the external symbol of the entry point in the 3704/3705 control program load module. (The standard entry for the Emulation Program is CYASTART.) If the SAVE option of the GEN3705 command is specified, this symbol is set in the output EXEC file according to the Stage 2 input file.

`NAME ncpname`
is the ncpname to be used when the DIAGNOSE parameter list is built. The ncpname specified must match an entry in the system name table. These entries are created with the NAMENCP macro when VM/SP is generated.

`LIBE libraryname`
is the filename of a load module library file, filetype LOADLIB, which contains the control program image as member 'fname'.

CAMOD $\begin{Bmatrix} 0 \\ 1 \end{Bmatrix}$

must be specified if a Type 4 Channel Adapter is being used. VM/SP supports only one Type 4 Channel Adapter at a time, although two may be present.

CAMOD 0 corresponds to -0 following the subchannel address on the ADDRESS operand of the LINE macro in Stage 1 of the EP system generation. (0 may have been coded or defaulted on the LINE macro; you must specify it on the CAMOD option.)

CAMOD 1 corresponds to -1 following the subchannel address on the ADDRESS operand of the LINE macro in Stage 1 of the EP system generation.

Execution of the SAVENCP Program: The DIAGNOSE X'50' instruction invokes the CP module DMKSNC to:

- Interpret the parameter list (CCPARM) built by SAVENCP

- Check the parameter specifications against the NAMENCP macro for the 3704/3705 control program

- Write the page-format image·of the control program onto the appropriate CP-owned volume.

The parameter list for the DIAGNOSE instruction must start on a 4096-byte boundary.

When the DIAGNOSE X'50' instruction is executed, the module DMKSNC searches the DMKSNT module for a NAMENCP macro with the same ncpname as the one in the CCPARM parameter list. The values specified in the parameter list are compared to those specified in the NAMENCP macro. If any parameters conflict, an error message is displayed at the terminal. If no error conditions are detected, DMKSNC starts to transfer the control program image from CMS virtual storage to the CP-owned volume specified in the NAMENCP macro. Successful completion of this process completes the generation of a 3704/3705 control program for VM/SP use.

Step 10. Load the 3704/3705 Control Program

The 3704/3705 control program is automatically loaded each time the VM/SP system is loaded, if the CPNAME operand was specified on the RDEVICE macro when VM/SP was generated and if the 3704/3705 is online. If the CPNAME operand was not coded, you must issue the CP NETWORK LOAD command line to load a 3704/3705 control program into the 3704/3705 Communications Controllers' storage.

The NETWORK LOAD Command Line: Use the NETWORK LOAD command to initiate the loading of an EP control program into a 3704/3705 Communications Controller.

The format of the NETWORK LOAD command line is:

NETwork	LOAD raddr ncpname

where:

LOAD initiates the control program load operation.

raddr is the real address of the 3704/3705 to be loaded.

ncpname is the name, defined by a NAMENCP macro, of the 3704/3705 control program image to be loaded into the 3704/3705 specified by raddr.

***Execution of the NETWORK LOAD Command*:** The NETWORK LOAD command accesses the control program image using the information in the system name table (DMKSNT) entry created by the NAMENCP macro. If the 3704/3705 specified in the command is not in an "IPL Required" state at the time the command is issued, the message:

```
DMKNET461R CTLR raddr IPL NOT REQUIRED; ENTER "YES" TO CONTINUE:
```

appears at the terminal. If the reply to the message is other than "yes," the command terminates without loading the 3704/3705. Otherwise, the loader bootstrap routines are written to the 3704/3705 and loading starts. VM/SP does not execute the "bring-up" test routines as a part of the load process. If these tests are to be made, they must be run from a virtual machine with the 3704/3705 dedicated.

When the load of the control program image is complete, the command processor verifies that the 3704/3705 configuration described by the control program can be serviced by the VM/SP CP control blocks in storage.

***Special Considerations for Loading the EP 3704/3705 Control Program*:** If a 3704/3705 Emulation Program is automatically reloaded after a 3704/3705 failure, the system may loop after the restart. The message

```
    DMKRNH463I CTLR raddr UNIT CHECK; RESTART IN PROGRESS
```

and two responses

```
    CTLR raddr DUMP COMPLETE
    CTLR raddr ncpname LOAD COMPLETE
```

indicate that the 3704/3705 has been reloaded. If the system loops after the second response, you must reset all emulator lines from the 3704/3705 control panel.

If the automatic dump feature is not enabled, one of the messages

```
    DMKRNH462I CTLR raddr UNIT CHECK; IPL REQUIRED

    -- or --

    DMKRNH464I CTLR 'raddr' CC=3; DEPRESS 370X "LOAD" BUTTON
```

indicates a 3704/3705 abnormal termination. The 3704/3705 Emulation Program must be reloaded via the NETWORK LOAD command. If the system loops when an attempt is made to enable the lines, you must reset all emulator lines from the 3704/3705 control panel.

The procedure for resetting emulator lines from the 3704/3705 control panel is described in the *Guide to Using the IBM 3704 Communications Controller Control Panel* and the *Guide to Using the IBM 3705 Control Panel.*

Step 11. Logging on Through the 3704/3705

Because a 3704/3705 can support emulator-mode lines and can also support a variety of terminals, the procedure for logging on is sometimes complicated. Use the following procedure to log on to VM/SP.

1. Turn the power on for your terminal and wait 15 to 30 seconds.

2. Look for an online message at your terminal.

   a. If one of the following messages

   ```
   VM/370 online      xxxxxx xxxxxx

                 -- or --

   xxxxxx xxxxxx      VM/370 online
   ```

   appears at your terminal, your terminal is a 2741 connected to VM/SP via a 2701/2702/2703 line or via a 3704/3705 line in emulation mode. You can proceed with the normal logon procedure for your type of terminal, as described in the *VM/SP Terminal Reference.*

   b. If the message

   ```
   VM/370 online
   ```

   appears at your terminal, your terminal is a 1050, 3101, or CPT-TWX Model 33/35 terminal connected to VM/SP in EP mode. You can proceed with the normal logon procedure for your terminal type. This procedure is described in the *VM/SP Operator's Guide.*

Step 12. Applying PTFs to the 3704/3705 Load Library

If necessary, it is possible to apply Program Temporary Fixes (PTFs) directly to the 3704/3705 load library. The CMS ZAP program applies the PTF. See the *VM/SP Operator's Guide* for information on using the ZAP service program.

Testing the 3704/3705 Control Program

After you have generated a 3704/3705 control program, loaded it, and logged on, you may want to test the 3704/3705 control program. Several CP commands are provided to control the operation, check the status, and dump the contents of the 3704/3705. The NETWORK command loads and dumps any 3704/3705 control program. The existing CP commands (ENABLE, QUERY, DISPLAY, and VARY) also provide support for EP 3704/3705 control programs. The NCPDUMP command formats and prints a dump of 3704/3705 storage. Use these commands to test the 3704/3705 control program.

The NETWORK, ENABLE, DISABLE, NCPDUMP, QUERY, and VARY commands are described in the *VM/SP Operator's Guide* and the *VM/SP CP Command Reference for General Users.*

Appendix E. Example of Alternate CMS Nucleus Placement

This section demonstrates how to have your CMS system saved at a storage location other than the "standard" one provided by the product tape sample. You might want such an alternative for users with large storage requirements, e.g. using APL with large workspaces.

An alternate saved system example labeled CMSL has been included in the DMKSNT samples. The difference between the two CMS samples is the amount of real storage used for segment tables. The CMSL sample is stored at a 16 Meg address and requires 1024 bytes of real storage per CMS user. The CMS sample is stored below 2 Meg and only uses 128 bytes of real storage for each CMS user for segment table entries. The trade-off between real storage availability and CMS usage should be evaluated prior to CMS relocation.

Alternate CMS Nucleus Placement

The following procedure demonstrates how to relocate your CMS saved system for loading in an 8 Meg virtual machine. The segment will be loaded at X'790000'.

In order to relocate your CMS saved system, you need to:

- Change the SLC name in the CMSLOAD EXEC.
- Create an SLC file to contain the new load address location.
- Modify the DMKSNT entries for CMS.

General Procedure

1. Using the MAINT logon:

2. ACCESS 190 A

3. Modify the load list, 'CMSLOAD EXEC A'. The only differences will be the Set Location Counter (SLC) values used to determine where the shared portion of CMS will be loaded into storage. In the following example "SLC L190000" is the name of a CMS file which contains the address of the starting location for loading the main portion of the nucleus. This SLC statement precedes the DMSALP entry in the load list. "SLC L200000" marks the end of the CMS nucleus and it precedes the DMSOME entry. *Note:* This file may be changed by service and should be checked/repeated prior to a CMS rebuild after applying a PUT tape.

```
xedit cmsload exec a
set case upper
top
change / SLC L190000 / SLC L790000 /*
top
change / SLC L200000 / SLC L800000 /*
file
```

4. Create two new SLC files to match the new load list:

```
xedit slc l790000 a
input $SLC  790000
set hex on
change /$/X'02'/         ( X'02' is an unprintable
file                     loader control character.)

xedit slc l800000 a
input $SLC  800000
set hex on
change /$/X'02'/         ( X'02' is an unprintable
file                     loader control character.)
```

Note: On the preceding "input" commands, there must be two blanks between SLC and the address.

5. Modify the SYSPGNM and SYSHRSG entries for the CMS entry in the current DMKSNT.

```
******************************************************
*           AN EXAMPLE FOR 3330 DASD DEVICES
******************************************************
*  This CMS example is located just below 8M in storage
*    CYL 01, PAGE 01 to CYL 03, PAGE 24 (138 pages)
*    137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*
CMS  NAMESYS   SYSNAME=CMS,
               SYSVOL=VMSRES,
               SYSSTRT=(001,1)
               SYSPGNM=(0-4,14-33,1936-2047),
               SYSPGCT=137,
               SYSHRSG=(121,122,123,124,125,126,127),
               SYSSIZE=256K,
               VSYSADR=190,
               SYSCYL=98,
               VSYSRES=VMPK01
*
```

Figure 24. Example of Alternate CMS Nucleus Placement for 3330 DASD Devices

Appendix F. Example of Additional CMS Segment Containing Y-STATS

The procedure you need to generate a CMS system including large S- and Y-disk directories is described in this appendix. Since CMS is designed to run as a shared system, performance is generally improved when the S-STAT and Y-STAT are included in the CMS nucleus.

If, when IPLing CMS, you see the message:

```
DMSACC723I Y (19E) R/O
```

the Y-STAT could not fit within the CMS nucleus at the time the system was saved. Every user who IPLs CMS gets the Y-disk accessed in non-shared user free storage.

Modification of CMS Nucleus to Include Large S- and Y-STATS

You can enlarge the CMS nucleus to contain the Y-STAT. In the procedure that follows, you:

- Determine the number of additional segments needed.
- Alter the CMS loadlist (CMSLOAD EXEC).
- Change the DMKSNT entry for CMS (and possibly other segments).
- Create an SLC Lxxxxxx file referenced by the CMSLOAD EXEC.

Note: A standard CMSLOAD EXEC as shipped with your system is assumed for the following procedure.

General Procedure

Step 1. Compute the New CMS Ending Address

To determine the new CMS ending address (NUCOMEGA), complete the following:

- Obtain the number of files (F) on both the S- and Y-disks. Issue:

  ```
  query disk      record the number of files on both disks.
  ```

- Record the hexadecimal addresses for DMSSIG and DMSOME. Compute the difference (D) between them and convert D to a decimal value.

- Compute the additional amount of storage needed (A) by using the applicable formula below:

  A = 40 * F - D (for 800-byte formatted disk)

  - or -

  A = 64 * F - D (for all others)

  where F is the number of files on S- and Y-disks and D is address difference in bytes.

- Use the table below to determine the increment to be added to the old CMS ending address.

If A is more than	but less than or =	needed segments	address increment
00000	65536	1	X'10000'
65537	131072	2	X'20000'
131073	196608	3	X'30000'

- Determine the new CMS ending address (NUCOMEGA) by adding the address increment hexadecimal value from above to X'200000'.

Note: This method of computing address increments yields approximate values. It does not take into account variables such as doubleword boundaries or headers. Therefore, you may need to make an adjustment to the next higher segment to compensate. Also, you may have to recompute the CMS ending address each time you apply service.

Step 2. Modify the CMSLOAD EXEC.

Use the System Product Editor to change the CMSLOAD EXEC. Issue the following commands:

```
xedit cmsload exec
top
c/ SLC L200000 / SLC Lxxxxxx / *
file
```

where "xxxxxx" is newly computed CMS ending address (NUCOMEGA).

Step 3. Modify the DMKSNT entry for CMS

To make the necessary changes in the DMKSNT file, issue:

```
xedit dmksnt assemble
```

There are three parameters on the CMS NAMESYS macro that will have to be changed.

- SYSSTRT

  Alter this parameter to point to an area on disk that can hold the CMS system now that its size has increased.

- SYSPGNM

  Increase the third group of pages to cover all of the extra segments to be shared.

- SYSHRSG

  Increase this parameter to include the extra segments to be shared.

Note: The above modifications may require a repositioning of various other segments defined in DMKSNT.

Step 4. Create an SLC Lxxxxxx file.

To create a new SLC Lxxxxxx file that is referenced by CMS, issue:

```
acc 190 b/a
copyfile slc l200000 b = lxxxxxx a
xedit slc lxxxxxx a
top
c / 200000 / xxxxxx / *
file
```

where "xxxxxx" is newly computed CMS ending address (NUCOMEGA).

After these four steps have been done, perform the standard CP and CMS nucleus generation procedures.

Appendix G. Special Options for CP

Examples of the output from the GENERATE EXEC when the Virtual=Real (V=R) Option or the Small CP Option is selected are illustrated below. These options require significant system programming experience. You should read and understand "Part 1. Planning for System Generation" of the *VM/SP Planning Guide and Reference* prior to selecting these options.

Small CP Option

Invoking the Small CP Option removes V=R support, AP/MP support, and support for:

SUPPORT	MODULES
3066	DMKGRH
Remote 3270	DMKRGA, DMKRGB, DMKRGC,DMKRGD
3340 Alternate Track	DMKTRK
3375/3380	DMKDAD
3704/3705	DMKRNH
3800 Printers	DMKVSV
3850 MSS	DMKSSS, DMKSSU
Missing Interrupt	DMKDID
MVS Guest	DMKFPS, DMKQVM, DMKVSC
SNA (CCS)	DMKVCP, DMKVCR, DMKVCT,DMKVCV, DMKVCX
TTY Terminal Support	DMKTTZ

If you need any of the support listed above, you must reply **no** to **DO YOU WANT THE SMALL CP OPTION?**. You may respond to the prompt in "Chapter 2. Installation When Using the Starter System Tape" or in "Chapter 3. Installation When Not Using the Starter System Tape." You may also begin this procedure at CP nucleus generation by entering:

generate cp nucleus

In either case, after you respond to the Small CP prompt, the procedure continues:

```
'194' REPLACES ' A (194) '
DO YOU WANT THE SMALL CP OPTION? -- ENTER:  ( YES | NO )
```

yes

```
READER EMPTY
NO  FILES PURGED

VMFLOAD BEING EXECUTED
LOADLIST= CPLOADSM AND CNTRL= DMKSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn  TO  MAINT    COPY 001   NOHOLD

AN IPL'ABLE CP NUCLEUS EXISTS IN YOUR VIRTUAL CARD READER.
DO YOU WISH TO CREATE .........
```

Virtual=Real Option

Be sure that there is enough storage for the CP nucleus with a virtual=real area before you load it. You can load a CP nucleus that has a virtual=real area in either a real or virtual machine. IPL the tape containing the CP nucleus. For the MAINT virtual machine, it is on the tape at virtual address 182.

To load a CP nucleus that has a virtual=real area, you must reply yes when asked **VIRTUAL=REAL OPTION REQUIRED (YES | NO)**, in "Chapter 2. Installation When Using the Starter System Tape" or in "Chapter 3. Installation When Not Using the Starter System Tape." You may also begin this procedure at CP nucleus generation by issuing:

```
  generate cp nucleus
```

In either case, after you respond to the V=R prompt, the procedure continues:

```
STORAGE SIZE OF VIRT=REAL  {MINIMUM IS 32K}:

   32k

00032K STORAGE SIZE FOR VIRTUAL=REAL
IS THE ABOVE ENTRY CORRECT (YES|NO):

   yes

READER EMPTY
NO  FILES PURGED

VMFLOAD BEING EXECUTED
LOADLIST= VRLOAD AND CNTRL= DMKSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn  TO  MAINT  COPY 001  NOHOLD

AN IPL'ABLE CP NUCLEUS EXISTS IN YOUR VIRTUAL CARD READER.
DO YOU WISH TO CREATE AN IPL'ABLE TAPE (FOR BACK-UP),
OR LOAD THE NUCLEUS IN YOUR READER DIRECTLY
TO YOUR SYSTEM RESIDENCE DISK VOLUME?
ENTER: ( TAPE | DISK  EXIT )

   tape

IPLABLE NUCLEUS NOW ON TAPE ****

THE NUCLEUS LOAD MAP MUST BE SAVED ON DISK.
WHEN YOU RECEIVE THE MESSAGE: NUCLEUS LOADED ON 'volid',
ENTER: 'CLOSE PRT' then 'IPL 190 PARM AUTOCR'
TO READ THIS LOAD MAP TO YOUR 194 DISK
ENTER: 'GENERATE MAP'

SINCE YOU HAVE CHOSEN THE V=R OPTION, DEFINE STORAGE LARGE ENOUGH
TO HOLD THE LOADER AND NUCLEUS, INCLUDING THE V=R AREA.

TO LOAD THE CP NUCLEUS JUST CREATED, IPL THE TAPE ( IPL 182 CLEAR ),
OR YOUR VIRTUAL CARD READER ( IPL 00C CLEAR ).
ONCE THE NUCLEUS HAS BEEN LOADED, READ IN THE LOAD MAP, AS DESCRIBED
ABOVE. THEN YOU MAY SHUTDOWN THE SYSTEM AND IPL YOUR NEWLY
CREATED CP SYSTEM.
R;
```

Appendix H. Installation Reference Worksheet

Record important instructions

Instruction **Number** **Address** **Name (Label)**

Mount volume ________________ as ____________ ________________________

Mount volume ________________ as ____________ ________________________

Mount volume ________________ as ____________ ________________________

Mount volume ________________ as ____________ ________________________

Mount tape ________________ as ____________ ________________________

Mount tape ________________ as ____________ ________________________

Record device addresses and device types

Device	**Contents (Label)**	**Address**	**Device Type**
System volume	System residence (VMSRES)		
Work volume	Starter system (VMPK01)		
Extra work volume	Device dependent (VMSEXT)		
Storage diskpack	Source code (VMSTGE)		
Storage diskpack	Source code (VMSTG2)		

Record other device addresses and device types

Device	**Address**	**Device Type**
Console		
First Tape Drive		
Second Tape Drive		
Printer		
Punch		
Reader		
Graphics		

Appendix I. Optional Feature Program Products

Optional Feature Program Products

Following is a list of the optional feature program products that are supported with VM/SP 3.1. Also shown are any prerequisite products required to install or use the products.

PRODUCT	ABBREV	NUMBER	LEVEL	PREREQ
A Departmental Reporting System II	ADRS	5796-PLN	2.1.8	5748-AP1
ADRS-II Business Graphics Feature	ADRS BG		2.1.8	5748-XXH
APL Data Interface II	APL/DI	5796-PNG	2.1.5	5748-AP1 5734-LM5
Cross System Product/Application Development	CSP/AD	5668944	1.1.1	5746-AM2 5768-945
Cross System Product/Application Execution	CSP/AE	5668945	1.1.1	5746-AM2
Directory Maintenance	DIRMAINT	5748-XE4	1.2.0	
Display Management System/CMS	DMS/CMS	5748-XXB	1.2.0	
Document Composition Facility	DCF	5748-XX9	1.3.0	
370X Emulator Program	EP/VS	5744-AN1	1.3.0	
Environmental Record Editing and Printing[1]	EREP	5749-010	2.3.0	
Financial Planning System II	FPS	5798-DCN	2.1.3	5748-AP1
Fortran Interactive Debug	FORTRAN DEBUG	5734-FO5	1.2.2	
Graphical Data Display Manager with Presentation Graphics and Interactive Map Definition Features	GDDM/PGF/IMD	5748-XXH	1.3.0	
High Performance Option[2]	HPO	5664-173	1.3.2	
IBM BASIC		5668-996	1.1.1	
Information Systems-VM & Information/VM-VSE Data Feature[3]	INFO/SYS	5735-OZS	1.2.0	5746-AM2

Figure 25 (Part 1 of 3). VM/SP Optional Feature Program Products

PRODUCT	ABBREV	NUMBER	LEVEL	PREREQ
Interactive Instructional Presentation System	IIPS	5668-012	1.2.0	
Interactive System Productivity Facility	ISPF	5668-960	1.1.0	
ISPF/Program Development Facility	ISPF/PDF	5664-172	1.1.0	5668-960
OS COBOL Interactive Debug	COBOL DEBUG	5734-CB4	1.1.4	
OS COBOL/VS	COBOL	5740-CB1	1.2.4	
OS PL/I Optimizer	PL/I OPT.	5734-PL1	1.4.0	
OS PL/I Optimizer and Libraries	PL/I OPT./LIB.	5734-PL3	1.4.0	
OS PL/I Resident Library	PL/I RES	5734-LM4	1.4.0	
OS PL/I Transient Library	PL/I TRANS	5734-LM5	1.4.0	
Professional Office System Program Product[4]	PROFS	5664-176		5748-XX9
Remote Spooling Communications Subsystem Networking	RSCS	5748-XP1	1.3.0	
Structured Query Language/Data System	SQL/DS	5748-XXJ	1.2.0	
Virtual Storage Extension/Virtual Storage Access Method	VSE/VSAM	5746-AM2	1.3.0	
VM BATCH Subsystem	BATCH	5796-BCY	1.1.2	
VM File Storage Facility	VMFSF	5798-DMY	1.1.0	
VM Real Time Monitor	SMART	5796-PNA	1.1.3	
VM/Interactive File Sharing	VM/IFS	5748-XXC	1.1.0	5746-AM2
VM/Interactive Problem Control System Extended	IPCS/E	5748-SA1	1.2.0	
VM/Interactive Productivity Facility	VM/IPF	5748-MS1	1.5.0	5668-960
VM/Pass-Through Facility	PVM	5748-RC1	1.2.0	
VM/Performance Monitor Analysis	VMAP	5798-CPX	1.3.7	5734-LM5

Figure 25 (Part 2 of 3). VM/SP Optional Feature Program Products

PRODUCT	ABBREV	NUMBER	LEVEL	PREREQ
VS APL	VSAPL	5748-AP1	1.4.0	5748-XXH[5]
VS FORTRAN	FORTRAN	5748-FO3	1.3.0	

Figure 25 (Part 3 of 3). VM/SP Optional Feature Program Products

[1] EREP is ordered as a feature of the SCP 5749-010.

[2] HPO is ordered separately as Program Product number 5664-173.

[3] The VM/VSE Data Feature is available as Program Product number 5668-919. The MVS Data Feature is available as Program Product number 5665-955. Ordering information for either of these products can be found in the latest announcement letters for these products. The Packaging Option registration process is not used for ordering these two products.

[4] PROFS, Program Product number 5664-176, installation is supported. To order PROFS and the associated documentation, refer to the latest announcement letter. The Packaging Option registration process is not used for ordering PROFS.

[5] Required only when using the VS APL Session Manager (AP126).

Reference Manuals

There are reference manuals for all of the Feature Products in the above chart. Refer to the VM/SP 3.1 ordering instructions for the list of available feature program product manuals. Your IBM representative or nearest IBM branch office can help you obtain these publications and can tell you how to subscribe to them so you automatically get updates and new editions.

DASD Requirements

Before you install your program products, you will want to ensure that MAINT 319 (the P-disk) and MAINT 325 (the W-disk) are large enough for each DASD type as determined by the combinations of products you choose to install. Figure 26 shows the number of 1K blocks required on the MAINT 319 and MAINT 325 minidisks for each program product. Figure 27 shows the 1K sizes of the MAINT 319 and MAINT 325 minidisks as defined by the sample directories for each DASD type. If your total of 1K blocks needed exceeds the space allocated in the sample directories, you will need to update your directory with larger sizes for the MAINT 319 and MAINT 325 minidisks as required.

Program Product	Product Unique Userid	Number of 1K Blocks on Common Program Product Minidisks 319	325
ADRS		5	1,135
ADRS BG		5	2,285
APL/DI		5	1,455
BATCH	VMBATCH VMBATCH1	75	
COBOL		2,450	
COBOL DEBUG		850	
CSP/AD	CSPUSER	15	
CSP/AE	CSPUSER	30	
DCF		1,860	
DIRMAINT	DIRMAINT DATAMOVR	485	
DMS/CMS		720	
FILE STORAGE	FSFTASK1 FSFCNTRL	270	
FORTRAN DEBUG		220	
EP/VS		5	
FPS		5	670
EREP		25	
GDDM/PGF/IMD		7,600	
IBM BASIC		15	
IIPS	IIPS	10	
INFO/SYS	ISMAINT	15	
IPCS/E	IPCS	725	

Figure 26 (Part 1 of 2). Feature Product DASD Utilization

Program Product	Product Unique Userid	Number of 1K Blocks on Common Program Product Minidisks 319	325
ISPF	ISPVM	5	
ISPF/PDF	ISPVM	5	
PL/I OPT		3,825	
PL/I OPT/LIBS		4,360	
PL/I RES		265	
PL/I TRANS		280	
PROFS	PRODBM PROMAIL SFBATCH SFCAL SYSADMIN	65	
PVM	PVM	40	
RSCS	RSCS	60	
SMART	SMART	10	
SQL/DS	SQLDBA SQLUSER	10	
VM/IFS		480	
VM/IPF	ADMIN CPRM DISKACNT SYSDUMP1 VMUTIL	5	
VMAP	VMAP	5	
VS APL		3,175	
VS FORTRAN		1,310	
VSE/VSAM		1,935	

Figure 26 (Part 2 of 2). Feature Product DASD Utilization

The following notes apply to Figure 26:

Notes:

1. For additional information conncerning feature products which require more space than listed above, refer to "Considerations for Installing Optional Feature Program Products," in Chapter 6.

2. The minidisk addresses listed here and in the sample directory are the default addresses for their corresponding products. During installation, program products assume that the sample entries have not been modified.

DASD TYPE	MAINT 319	MAINT 325
3310	21,300	3,250
3330	18,810	4,180
3340	18,900	4,032
3350	20,250	4,050
3370	21,300	4,560
3375	19,500	4,500
3380	20,925	4,185

Figure 27. Size of Allocated Minidisks in 1K Blocks

DASD TYPE	1K BLOCKS/CYLINDER
3330	209
3340	84
3350	450
3375	300
3380	465

3310 and 3370 are defined in
512 byte blocks. Multiply
the 1K block requirements by
2 to determine the number of
512-byte blocks required.

Figure 28. 1K Blocks/Cylinder per DASD Type

Directory Entries

This section gives a brief description of the userids found in the sample files.

Disconnected Virtual Machines

- AUTOLOG1

  VM/SP automatically logs on this virtual machine when the VM/SP SYSTEM is IPLed. The PROFILE EXEC for this userid can therefore be used to automatically perform any actions that may be desired every time the system is IPLed. Typically, this includes automatically logging on any other disconnected virtual machines, enabling lines and/or starting printers, etc.

- CMSBATCH

  CMSBATCH is used to better utilize system and personnel resources. Processor intensive jobs may be routed to CMSBATCH, thus freeing the user's terminal for more interactive work and preventing processor intensive jobs from bogging down the system. For more information about the CMSBATCH virtual machine, refer to the *VM/SP CMS User's Guide* and the *VM/SP System Programmer's Guide.*

- DATAMOVR

  The DATAMOVR virtual machine is defined to automate the process of copying user CMS minidisks and cleaning old minidisks prior to making them available for subsequent allocation.

- DIRMAINT

  The DIRMAINT virtual machine owns the CP directory source and associated control files. DIRMAINT makes changes to the CP directory source so that users do not have to edit the file manually.

- ISPVM

  ISPVM is required to run ISPF. ISPVM is to be run in the disconnected state, and should be logged on at system IPL time using the AUTOLOG1 virtual machine.

- OPERATNS

  The OPERATNS userid is required by the VM/IPF release 5 Problem Control Facility (PCF). All IPCS Extended problems are sent to this userid, which provides standard IPCS Extended services. All CP system dumps are also sent to OPERATNS, due to the "SYSDUMP=OPERATNS" entry in the DMKSYS.

  Disks associated with the OPERATNS userid are as follows:

  OPERATNS 191 is used as a work disk to process dump files.
  OPERATNS 193 contains one current dump and all of the previous reports generated through use of the VM/IPF release 5 Problem Control Facility dialogs.

  For more information about the OPERATNS virtual machine, refer to the *VM/SP Planning Guide and Reference.*

- PVM

  VM/Pass-Through Facility runs in the PVM virtual machine.

  The VM/Pass-Through Facility licensed program provides VM display station users with the ability to interactively access the local VM system as well as remote systems. Remote systems can be other VM systems (with or without Pass-Through installed), or they can be non-VM systems.

- ROUTER

  The ROUTER disconnected virtual machine is used by the VM/VSE feature of the VSE/SIPO to route virtual spool files from a production/batch machine to specified CMS users.

  The ROUTER machine has the important function of supporting CMS users who require output from a production/batch virtual machine to be spooled back to their virtual reader. This function helps reduce output on system unit record devices, and provides the benefits associated with this (I.E. less paper wastage, less operator intervention, etc).

- RSCS

  RSCS runs in a single virtual machine as a telecommunications subsystem for the VM/SP system. It requires a standard VM/SP virtual machine with telecommunications facilities attached to that virtual machine. Normal operation can be conducted without operator intervention, and RSCS is run disconnected.

- SMART

  SMART is designated as a real-time monitor and diagnostic tool for short term monitoring, analysis, and problem solving. It is recommended that SMART be used for installations of hardware or software to assist in validating the system components and/or establish requirements for additional hardware or software.

- VMAP

  Those installations that wish to take advantage of the VM/SP performance monitoring capabilities for long-term capacity planning and system tuning should install the VM/370 Performance Monitor/Analysis FDP on the VMAP virtual machine to do the data reduction and report processing. This virtual machine is not automatically logged on.

  If you want to establish automatic data collection and data reduction, you should do the following:

  1. Install the VM/370 Performance Monitor/Analysis FDP on VMAP.

  2. Add the VMAP userid to the list of automatically logged-on virtual machines in the AUTOLOG1 virtual machine.

  3. Change the SYSMON macro in DMKSYS, setting the AUTO parameter to YES.

  4. Reassemble DMKSYS and generate a new CP nucleus.

- VMBATCH and VMBATCH1

  The VMBATCH and VMBATCH1 userids are used by the program product VM batch subsystem. This program provides a facility for scheduling, initiating, and monitoring batch jobs within a VM/CMS environment. The subsystem is controlled by a supervisory virtual machine which dispatches and monitors other virtual machines in which the batch jobs are processed.

  The VM batch subsystem monitor receives and responds to jobs and commands from users. When a batch machine becomes available to process a job, the monitor selects an appropriate batch job and starts the job in the machine.

Disconnected Virtual Machines Used by IPF

VM/IPF release 5 relies on the following userids running as disconnected virtual machines to provide frequently used services.

- CPRM

  The CPRM (Central Problem Report Manager) userid is required by the VM/IPF release 5 Problem Control Facility (PCF). This disconnected service machine has the authority to control and change the problem data base.

  Disks associated with the CPRM userid are as follows:

CPRM 191	is used as a work disk.
CPRM 192	contains problem reports submitted and updated by users, problem reports created as the result of a dump, and control files.
CPRM 291	contains files required for execution of the VM/IPF release 5 Problem Control Facility dialogs.

- DISKACNT

  DISKACNT provides for the collection of accounting information generated by VM/SP.

- SYSDUMP1

  The SYSDUMP1 virtual machine handles the DASD volume backups, either on a specified schedule, or as requested by an administrator or the operator. This virtual machine also provides services for restoring a specified user's CMS files from backup tapes.

- VMUTIL

  The VMUTIL virtual machine handles a variety of tasks that occur repeatedly, on a schedule selected by the installation. Some examples of these tasks might include:

  - Scheduling periodic DASD volume backups
  - Repeating a tape mount request periodically

Note: The following disconnected virtual machines are also used by IPF if the appropriate optionsl feature program products are installed:

AUTOLOG1
CMSBATCH
DATAMOVR
DIRMAINT
ISPVM
OPERATNS
RSCS

Other Userids

In addition to those userids that run as disconnected virtual machines, the following userids must be included in the system directory.

- ADMIN

  The ADMIN userid is reserved for the system administrator. The system administrator's tasks include authorizing users of the system and controlling the allocation of system resources (primarily DASD). These functions are supported by the VM/IPF release 5 Dirmaint, Problem Control Facility, and Tailoring dialogs.

- CMSUSER

  CMSUSER is provided as an example of a virtual machine required by the typical CMS user. Since you will probably have more than one user of CMS, it will be necessary to create similar entries for each user.

- CSPUSER

  The CSPUSER userid is used by the Cross System Product/Application Development (CSP/AD) and Cross System Product/Application Extension (CSP/AE) for interactive development application and extension.

- EREP

  EREP is for the use of the IBM Customer Engineer while running CPEREP. The privilege class of F allows the CE to clear the error recording area and the maximum virtual size is set at two megabytes so the EREP table can be increased in size if necessary. The minidisk associated with EREP is:

  EREP 191 may be used to save often-used EREP control statements and procedures (EXECs).

- FSFADMIN and FSFCNTRL

  The userids FSFADMIN and FSFCNTRL correspond to the VM File Storage Facility that provides disk space and data file management services for users of CMS. These services are available to users within a single host environment or within a multiple-system computing network. Data and disk space may be shared, on a voluntary basis, with varying levels of protection selected by individual users. A hierarchical file directory permits users to optionally cluster files in a logical manner, and to increase resolution in the identification of data files.

- IIPS

  The IIPS userid controls the instructional system invoked by VM/IPF release 5. Minidisks associated with the IIPS userid are as follows:

  IIPS 191 is used for workspace, registering students, etc.
  IIPS 193 contains on-line courses.

- IPCS

IPCS is the userid reserved for system support. This userid has a link to OPERATNS 193.

- ISMAINT

ISMAINT is the userid for the Information System administrator. Minidisks associated with ISMAINT are as follows:

ISMAINT 191 is used for the Info/System programs and EXECs.
ISMAINT 29x minidisks are used for the VSAM databases on which the Info/System Data Feature resides. The VSAM master catalog and the glossary VSAM cluster reside on 290. The text VSAM cluster begins on 291 and can be spread over as many as 5 minidisks (291-295). Access modes for 290-295 are G, F, E, D, C, and B respectively. These minidisks have not been included in the distributed Directories.

- LEV2VM

LEV2VM is the userid used for running a second level VM system (VM under VM). LEV2VM has no minidisks of its own, since VM under VM is normally done on a temporary basis only. Before using this machine you must attach the volumes required by your second level system to this userid.

- MAINT

The MAINT userid is traditionally used by the system programmer to perform system maintenance activities. These functions are supported by the VM/IPF release 5 System Support, RSCS, DIRMAINT, and tailoring dialogs. If the system programmer does other work on the system, establish another userid for this activity, reserving the MAINT userid for system updates and maintenance only.

This userid "owns" all of the system files. Disks associated with the MAINT userid are as follows:

MAINT 123 is used to format or backup the entire VMSRES volume and to re-write the CP nucleus.
MAINT 124 is used to format or backup the entire VMSEXT volume.
MAINT 125 is used to format or backup the entire VMPK01 volume.
MAINT 126 is used to format or backup the entire VMSTGE volume.
MAINT 190 is the CMS system disk. The CMS nucleus and all of the CMS TEXT decks are located here. MAINT 190 is linked by all CMS users as their 190 disk.
MAINT 191 is the workspace and location of tailored VM/SP files.
MAINT 194 is the location of TEXT decks and MACLIBs required to generate the CP nucleus and also UPDATE files for attached processor support.
MAINT 196 used with High Performance Option.
MAINT 19D is the location of files containing explanations of CP, CMS, and XEDIT commands that are used by the CMS HELP facility. MAINT 19D is linked by all CMS users as their 19D disk.

MAINT 19E is the CMS system extension disk. All of the VM/IPF release 5 General User execs and functions are kept here. MAINT 19E is linked by all CMS users as their 19E disk.

MAINT 201 used to hold the EREP library.

MAINT 293 is the location of AUX and update files used for CMS service.

MAINT 294 is the location of AUX and update files used for CP and RSCS service.

MAINT 296 used with High Performance Option.

MAINT 300 is the location of all of the execs and functions that belong to the VM/IPF release 5 System Support, Administration, and Operation dialogs.

MAINT 310 is the location of the MACLIBs that belong to VM/IPF release 5. These MACLIBs contain all of the VM/IPF release 5 panels and message line messages. MAINT 310 is also the location of the CLRSCRN MODULE, IPFFIPF DEFAULT2, and RGGLLIB LOADLIB files.

MAINT 319 is the location of the optional feature program products.

MAINT 31A is the location of the locally installed licensed programs and the locally written application programs.

MAINT 324 is the location of the IBM BASIC optional feature program product.

MAINT 325 is the location of the featured Field Developed Programs (FDPs) and Installed User Programs (IUPs).

MAINT 393 is the location of the source for the CMS portion of VM/SP.

MAINT 394 is the location of the source for the CP portion of VM/SP.

MAINT 396 used with High Performance Option.

MAINT 3A0 is the location of an online document named "IPF Module/EXEC Guide" which is shipped with VM/IPF release 5. Guides shipped with other program products are also kept here.

- OLTSEP

OLTSEP is for the use of the IBM Customer Engineer while running the OnLine Test System Executive Program (OLTSEP). The privilege class of F allows the CE to specify intensive recording mode. Options have been set as required by OLTSEP. The minidisk associated with OLTSEP is:

OLTSEP 5FF is the CE's OLTSEP volume, which will be automatically IPLed in the virtual machine.

- OPERATOR and OP1

The OPERATOR and OP1 userids are used by the operator to control the system. If the VM/IPF release 5 operation dialogs are run on the OPERATOR userid, any system and user messages that are sent to the operator will be displayed only when the user presses the ENTER key or otherwise clears the screen. To avoid these interruptions, you can log onto the OP1 userid at a terminal and run the operator dialogs from there. The OPERATOR userid can then be logged onto the console and reserved for system and user messages.

- OSVS1

  OSVS1 is the userid used by a guest OS/VS1 operating system. OSVS1 has no minidisks of its own. The Directory entry must be updated to supply access to the volumes required by your guest OS/VS1 system.

- VSEIPO

  VSEIPO is the userid used by the VSE operating system. VSEIPO owns the VSE volumes and has *read only* access to VSEMAINT's 191 minidisk. The VSEIPO userid entry in the supplied VM/SP directory has all minidisk statements entered as comments. Before using the VSEIPO userid, you must determine the DASD type that will be used for your VSE system and update the directory accordingly before using it.

- VSEMAINT

  VSEMAINT is the userid for performing all tailoring and service of the VSE system. Interactive Productivity Facility files will be created on the 191 minidisk when installing the VM/VSE feature of IPF. These files must be maintained for the VSE system management functions of the VM/VSE feature to perform correctly. This userid is supplied with no minidisks. Therefore, you must update the directory before using it. The VM Directory Maintenance Program Product may be used for this.

USERIDS for the PROFS System

- PRODBM

  The PRODBM disconnected virtual machine offers facilities for storing all documents and controlling their security. It is also used during user enrollment.

  The PRODBM minidisk assignments are as follows:

  PRODBM 191 A-disk, contains TEXT decks as well as a PROFILE EXEC to initiate the PROFS data base manager and other data base control files.
  PRODBM 161 B-disk, contains stored Index information for all documents on the system.
  PRODBM 5FF first document storage disk.
  PRODBM 5FE second document storage disk.
  PRODBM 5FD third document storage disk.

- PROMAIL

  The PROMAIL disconnected virtual machine offers facilities for delivery and receipt of all documents.

  The minidisks assigned to PROMAIL are used as follows:

  PROMAIL 191 A-disk, contains ZIPMAIL module, buck slips, and remote locations files as well as a PROFILE EXEC to initiate the PROFS distribution manager.

PROMAIL 151 contains information stored about incoming and outgoing documents.

- SFBATCH

  The Scheduling Facility Batch Printer prints conference room schedules and appointment calendars.

  SFBATCH minidisk usage follows:

  SFBATCH 191 A-disk, contains EXECs and MODULEs for the control of the machine as well as a PROFILE EXEC to initiate the PROFS Calendar Printing Virtual Machine.

- SFCAL

  The Scheduling Facility Data Base disconnected virtual machine stores and updates conference room schedules and appointment calendars. It is also used during user enrollment.

  SFCAL's minidisks are used as follows:

  SFCAL 191 A-disk, contains the PROFILE EXEC and the AA1SFDBM module, controlling the execution of this machine.
  SFCAL 196 storage disk for conference room schedules and appointment calendars.

- SYSADMIN

  SYSADMIN is the userid used by the System Administrator who is responsible for the maintenance and administration of the PROFS PP system.

  The minidisks assigned to the SYSADMIN userid are used as follows:

  SYSADMIN 191 A-disk, contains the PROFLOAD EXEC and OFSCPUID.
  SYSADMIN 398 contains PROFS TEXT decks, EXECS, and files used for the PROFS disconnected virtual machines and for loading PROFS in DCSS.
  SYSADMIN 399 contains PROFS user routines and supporting files.
  SYSADMIN 298 contains PROFS help screens, messages and supporting EXECS.

USERIDS for the SQL/DS System

- SQLDBA

  SQLDBA is the userid under which all SQL/DS data bases are generated, and is referred to as the SQL/DS data base machine.

  SQLDBA's minidisks are used as follows:

  SQLDBA 191 A-disk, work disk for the SQL/DS data base machine.
  SQLDBA 193 the SQL/DS service minidisk used for SQL/DS data base installation, generation, and maintenance activities.

SQLDBA 195	the SQL/DS production minidisk used for all SQL/DS operations.
SQLDBA 200	the directory disk for the SQL/DS starter data base.
SQLDBA 201	the log disk for the SQL/DS starter data base.
SQLDBA 202	the data disk for the SQL/DS starter data base.

- SQLUSER SQLUSER is a SQL/DS user machine used to to administrative tasks needed to complete installation.

  The SQLUSER SQLUSER's minidisk is used as follows:

SQLUSER 191	A-disk, work disk for the SQLUSER user machine.

Reclaiming DASD Space

The sample directories contain entries for each of the supported feature product userids. You may want to tailor these directories for better utilization of DASD space after the installation is complete. For instance, space allocated for products not installed may be used as a different userid or deleted. Refer to Figure 26 for unique userids for each of the products.

For example, if you did not install the FILE STORAGE program product, you can xedit the directory and remove the entries for userids FSFTASK1 and FSFCNTRL thereby reclaiming the minidisk space (cylinders [CKD] or blocks [FBA]) set aside for its unique disks that you would not be using.

Once you have made all your changes, use the CMS DIRECT command to process the file to see if it follows the required directory format.

```
DIRECT VMUSERS DIRECT (EDIT
```

Notes:

1. If DIRMAINT is installed, refer to the DIRMAINT documentation for directions for updating the directory.

2. You can also use the DISKMAP EXEC to produce a directory map.

Appendix J. 3310 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3310 Directory Supplied with the Product Tape

Sample Directory for a FB-512 3310 Device

```
***************************************************************
*  3310      SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 FB-512 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$ NOLOG
 MDISK A01 FB-512 000000 000016 VMSRES R
 MDISK B01 FB-512 000000 000016 VMSEXT R
 MDISK C01 FB-512 000000 000016 VMPK01 R
 MDISK C02 FB-512 000000 000016 VMPK02 R
 MDISK C03 FB-512 000000 000016 VMPK03 R
 MDISK D01 FB-512 000000 000016 VMSTGE R
 MDISK D02 FB-512 000000 000016 VMSTG2 R
 MDISK E01 FB-512 000000 000016 PROFPK R
 MDISK F01 FB-512 000000 000016 SQLPK  R
 MDISK F02 FB-512 000000 000016 SQLPK2 R
*
USER $TEMP$ NOLOG
 MDISK A06 FB-512 115716 006484 VMSRES R
 MDISK B06 FB-512 080248 020000 VMSEXT R
 MDISK C01 FB-512 027376 020000 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK A03 FB-512 091000 035016 VMPK01 R
*
```

Figure 29 (Part 1 of 13). Sample Directory for a 3310 Device

```
USER $CPNUC$ NOLOG
 MDISK A10 FB-512 123736 002280 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A04 FB-512 092480 001824 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 FB-512 000016 009360 VMSRES R
 MDISK C02 FB-512 000016 010450 VMPK01 R
 MDISK E01 FB-512 000016 001544 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A09 FB-512 122712 000512 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A07 FB-512 094304 000512 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 FB-512 122200 000512 VMSRES R
*
***************************************************************
*             SYSTEM RELATED USERIDS
***************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 053200 002280 VMPK02 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 030182 000192 VMPK03 MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 FB-512 025118 000960 VMPK03 MR RBATCH WBATCH MBATCH
*
```

Figure 29 (Part 2 of 13). Sample Directory for a 3310 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 020498 002280 VMPK01 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 FB-512 055480 000200 VMPK02 MR RCPRM WCPRM
 MDISK 192 FB-512 055680 006000 VMPK02 MR ALL WCPRM
 MDISK 291 FB-512 061680 000680 VMPK02 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 5
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 076960 004000 VMPK02 MR RCSP WCSP MCSP
 MDISK 193 FB-512 080960 005852 VMPK02 MR RCSP WCSP MCSP
 MDISK 502 FB-512 086812 018810 VMPK02 MR RCSP WCSP MCSP
 MDISK 503 FB-512 105622 018810 VMPK02 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 6 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 FB-512 023028 002090 VMPK03 M RMOVR WMOVR MMOVR
*
```

Figure 29 (Part 3 of 13). Sample Directory for a 3310 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 7 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 FB-512 022778 004560 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 193 FB-512 045448 007752 VMPK02 MR RDIRM WDIRM MDIRM
 MDISK 195 FB-512 015228 007800 VMPK03 MR RDIRM WDIRM MDIRM
 MDISK 123 FB-512 000000 126016 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 030374 000480 VMPK03 MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 FB-512 036202 000912 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT 88
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 FB-512 076944 000836 VMPK03 MR RFSF1 WFSF1
*
```

Figure 29 (Part 4 of 13). Sample Directory for a 3310 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT 99
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 FB-512 062732 006270 VMPK03 MR RFSF WFSF
 MDISK 192 FB-512 069002 002090 VMPK03 MR RFSF WFSF
 MDISK 193 FB-512 071092 002090 VMPK03 MR RFSF WFSF
 MDISK 194 FB-512 073182 001254 VMPK03 MR RFSF WFSF
 MDISK 195 FB-512 074436 001254 VMPK03 MR RFSF WFSF
 MDISK 197 FB-512 075690 000418 VMPK03 MR RFSF WFSF
 MDISK 200 FB-512 076108 000418 VMPK03 MR RFSF WFSF
 MDISK 400 FB-512 076526 000418 VMPK03 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT 10
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 000016 010776 VMPK02 MR ALL WIIPS MIIPS
 MDISK 193 FB-512 010792 016416 VMPK02 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 FB-512 000016 000456 VMPK03 MR RPSR WPSR MPSR
*
```

Figure 29 (Part 5 of 13). Sample Directory for a 3310 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 FB-512 043168 002280 VMPK02 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 027208 001330 VMPK02 MR RISPF WISPF MISPF
 MDISK 192 FB-512 028538 014630 VMPK02 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M   16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 123224 000256 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M   4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 123480 000256 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 29 (Part 6 of 13). Sample Directory for a 3310 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 FB-512 000000 126016 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 124 FB-512 000000 126016 VMSEXT MW RSYSRES WSYSRES MSYSRES
 MDISK 125 FB-512 000000 126016 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 FB-512 000000 126016 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 127 FB-512 000000 126016 VMPK02 MW RSYSRES WSYSRES MSYSRES
 MDISK 128 FB-512 000000 126016 PROFPK MW RSYSRES WSYSRES MSYSRES
 MDISK 129 FB-512 000000 126016 SQLPK  MW RSYSRES WSYSRES MSYSRES
 MDISK 130 FB-512 000000 126016 SQLPK2 MW RSYSRES WSYSRES MSYSRES
 MDISK 190 FB-512 046912 045568 VMSRES MW ALL     WMAINT  MMAINT
 MDISK 191 FB-512 009376 005282 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 194 FB-512 014658 020520 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 196 FB-512 100248 012540 VMSEXT MW RMAINT  WMAINT  MMAINT
 MDISK 19D FB-512 094816 020900 VMSRES MW ALL     WMAINT  MMAINT
 MDISK 19E FB-512 055156 025080 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 201 FB-512 047376 020520 VMPK01 MW RMAINT  WMAINT  MMAINT
 MDISK 293 FB-512 010466 010032 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 FB-512 074736 009614 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 FB-512 000016 012540 VMSEXT MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 FB-512 001624 003800 VMPK03 MW RMAINT  WMAINT  MMAINT
 MDISK 310 FB-512 062360 014600 VMPK02 MW ALL     WMAINT  MMAINT
 MDISK 319 FB-512 012556 042600 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 31A FB-512 006448 002280 VMPK03 MW ALL     WMAINT  MMAINT
 MDISK 324 FB-512 077780 012540 VMPK03 MW ALL     WMAINT  MMAINT
 MDISK 325 FB-512 008728 006500 VMPK03 MW ALL     WMAINT  MMAINT
 MDISK 390 FB-512 000472 001152 VMPK03 MW ALL     WMAINT  MMAINT
 MDISK 393 FB-512 000032 068166 VMSTGE WR RMAINT  WMAINT
 MDISK 394 FB-512 000032 071064 VMSTG2 WR RMAINT  WMAINT
 MDISK 396 FB-512 068198 029260 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 FB-512 084350 006400 VMPK01 MW ALL     WMAINT  MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF FB-512 000000 126016 CEPACK MR READ WRITE
*
```

Figure 29 (Part 7 of 13). Sample Directory for a 3310 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 005424 001024 VMPK03 MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 037114 000456 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 FB-512 115392 007296 VMSEXT MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 035178 001024 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT 250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 054978 002508 PROFPK MR RDBM WDBM MDBM
 MDISK 161 FB-512 057486 008360 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD FB-512 088418 010032 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE FB-512 078386 010032 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF FB-512 068354 010032 PROFPK MR RDBM WDBM MDBM
*
```

Figure 29 (Part 8 of 13). Sample Directory for a 3310 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT 250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM 191 395 RR
 MDISK 191 FB-512 105974 006270 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 FB-512 065846 002508 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 030854 002200 VMPK03 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 FB-512 XXXXXX XXXXXX VMXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 191 FB-512 072456 002280 VMPK01 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 113498 002508 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 29 (Part 9 of 13). Sample Directory for a 3310 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 112244 001254 PROFPK MR RCAL WCAL MCAL
 MDISK 196 FB-512 098450 007524 PROFPK MR RCAL WCAL MCAL
*
************************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
************************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT 25
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 FB-512 033054 006688 VMPK03 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT 26
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 000016 008100 SQLPK  W
 MDISK 193 FB-512 008116 018630 SQLPK  R  RSQL WSQL
 MDISK 195 FB-512 026746 009720 SQLPK  RR RSQL WSQL MSQL
 MDISK 200 FB-512 036466 023490 SQLPK  R  RSQL WSQL
 MDISK 201 FB-512 059956 009600 SQLPK  R  RSQL WSQL
 MDISK 202 FB-512 000016 092160 SQLPK2 R  RSQL WSQL
*
USER SQLUSER SQLUSER  2M 2M G
 ACCOUNT 27
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 FB-512 092176 001620 SQLPK2 W
*
```

Figure 29 (Part 10 of 13). Sample Directory for a 3310 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT 250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 FB-512 001560 004000 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 FB-512 005560 008980 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 FB-512 033478 009500 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 FB-512 042978 012000 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 026078 002280 VMPK03 MR RDUMP WDUMP MDUMP
 MDISK 123 FB-512 000000 126016 VMSRES RR
 MDISK 124 FB-512 000000 126016 VMSEXT RR
 MDISK 125 FB-512 000000 126016 VMPK01 RR
 MDISK 126 FB-512 000000 126016 VMSTGE RR
 MDISK 127 FB-512 000000 126016 VMPK02 RR
 MDISK 128 FB-512 000000 126016 PROFPK RR
 MDISK 129 FB-512 000000 126016 VMPK03 RR
 MDISK 130 FB-512 000000 126016 SQLPK  RR
 MDISK 131 FB-512 000000 126016 SQLPK2 RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 067896 004560 VMPK01 MR RVMAP WVMAP MVMAP
*
```

Figure 29 (Part 11 of 13). Sample Directory for a 3310 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT 18
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 039742 006270 VMPK03 W RBATCH WBATCH
 MDISK 192 FB-512 046012 008360 VMPK03 W RBATCH WBATCH
 MDISK 193 FB-512 054372 006270 VMPK03 W RBATCH WBATCH
 MDISK 194 FB-512 060642 000418 VMPK03 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT 19
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 FB-512 061060 001672 VMPK03 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 028358 001824 VMPK03 MR RUTIL WUTIL MUTIL
*
```

Figure 29 (Part 12 of 13). Sample Directory for a 3310 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3330 SYSTEM
*MDISK 150 3330 000 411 DOSRES MR VSEIPO VSEIPO
*MDISK 151 3330 000 411 SYSWK1 MR VSEIPO VSEIPO
*MDISK 152 3330 000 411 SYSWK2 MR VSEIPO VSEIPO
*MDISK 153 3330 000 411 SYSWK3 MR VSEIPO VSEIPO
*MDISK 154 3330 000 411 SYSWK4 MR VSEIPO VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO VSEIPO
* 3350 SYSTEM
*MDISK 350 3350 000 560 DOSRES MR VSEIPO VSEIPO
*MDISK 351 3350 000 560 SYSWK1 MR VSEIPO VSEIPO
*MDISK 352 3350 000 560 SYSWK2 MR VSEIPO VSEIPO
*MDISK 353 3350 000 560 SYSWK3 MR VSEIPO VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 000000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 000000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 000000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 29 (Part 13 of 13). Sample Directory for a 3310 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	FB-512	OLTSEP	5FF	MR	000	126015	126016	
PROFPK	FB-512	MAINT	128	MW	000	126015	126016	
		SYSDUMP1	128	RR	000	126015	126016	
		$ALLOC$	E01	R	000	015	016	
		$SAVSYS$	E01	R	016	1559	1544	
		SYSADMIN	191	MR	1560	5559	4000	
		SYSADMIN	298	MR	5560	14539	8980	
					14540	33477	18938	GAP
		SYSADMIN	398	MR	33478	42977	9500	
		SYSADMIN	399	MR	42978	54977	12000	
		PRODBM	191	MR	54978	57485	2508	
		PRODBM	161	MR	57486	65845	8360	
		PROMAIL	151	MR	65846	68353	2508	
		PRODBM	5FF	MR	68354	78385	10032	
		PRODBM	5FE	MR	78386	88417	10032	
		PRODBM	5FD	MR	88418	98449	10032	
		SFCAL	196	MR	98450	105973	7524	
		PROMAIL	191	MR	105974	112243	6270	
		SFCAL	191	MR	112244	113497	1254	
		SFBATCH	191	MR	113498	116005	2508	
					116006	126015	10010	GAP
SQLPK	FB-512	MAINT	129	MW	000	126015	126016	
		SYSDUMP1	130	RR	000	126015	126016	
		$ALLOC$	F01	R	000	015	016	
		SQLDBA	191	W	016	8115	8100	
		SQLDBA	193	R	8116	26745	18630	
		SQLDBA	195	RR	26746	36465	9720	
		SQLDBA	200	R	36466	59955	23490	
		SQLDBA	201	R	59956	69555	9600	
					69556	126015	56460	GAP
SQLPK2	FB-512	MAINT	130	MW	000	126015	126016	
		SYSDUMP1	131	RR	000	126015	126016	
		$ALLOC$	F02	R	000	015	016	
		SQLDBA	202	R	016	92175	92160	
		SQLUSER	191	W	92176	93795	1620	
					93796	126015	32220	GAP
VMPK01	FB-512	MAINT	125	MW	000	126015	126016	
		SYSDUMP1	125	RR	000	126015	126016	
		$ALLOC$	C01	R	000	015	016	
		$SAVSYS$	C02	R	016	10465	10450	
		MAINT	293	MW	10466	20497	10032	
		CMSUSER	191	MR	20498	22777	2280	
		DIRMAINT	191	MR	22778	27337	4560	
					27338	27375	038	GAP
		$TEMP$	C01	R	27376	47375	20000	
		MAINT	201	MW	47376	67895	20520	
		VMAP	191	MR	67896	72455	4560	
		RSCS	191	MR	72456	74735	2280	
		MAINT	294	MW	74736	84349	9614	
		MAINT	3A0	MW	84350	90749	6400	
					90750	90999	250	GAP
		$TDISK$	A03	R	91000	126015	35016	

Figure 30 (Part 1 of 3). Directory Map Sample for a 3310 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK02	FB-512	MAINT	127	MW	000	126015	126016	
		SYSDUMP1	127	RR	000	126015	126016	
		$ALLOC$	C02	R	000	015	016	
		IIPS	191	MR	016	10791	10776	
		IIPS	193	MR	10792	27207	16416	
		ISPVM	191	MR	27208	28537	1330	
		ISPVM	192	MR	28538	43167	14630	
		ISMAINT	191	MR	43168	45447	2280	
		DIRMAINT	193	MR	45448	53199	7752	
		ADMIN	191	MR	53200	55479	2280	
		CPRM	191	MR	55480	55679	200	
		CPRM	192	MR	55680	61679	6000	
		CPRM	291	MR	61680	62359	680	
		MAINT	310	MW	62360	76959	14600	
		CSPUSER	191	MR	76960	80959	4000	
		CSPUSER	193	MR	80960	86811	5852	
		CSPUSER	502	MR	86812	105621	18810	
		CSPUSER	503	MR	105622	124431	18810	
					124432	126015	1584	GAP
VMPK03	FB-512	SYSDUMP1	129	RR	000	126015	126016	
		$ALLOC$	C03	R	000	015	016	
		IPCS	191	MR	016	471	456	
		MAINT	390	MW	472	1623	1152	
		MAINT	300	MW	1624	5423	3800	
		OP1	191	MR	5424	6447	1024	
		MAINT	31A	MW	6448	8727	2280	
		MAINT	325	MW	8728	15227	6500	
		DIRMAINT	195	MR	15228	23027	7800	
		DATAMOVR	191	M	23028	25117	2090	
		CMSBATCH	195	MR	25118	26077	960	
		SYSDUMP1	191	MR	26078	28357	2280	
		VMUTIL	191	MR	28358	30181	1824	
		AUTOLOG1	191	MR	30182	30373	192	
		DISKACNT	191	MR	30374	30853	480	
		PVM	191	MR	30854	33053	2200	
		SMART	191	MR	33054	39741	6688	
		VMBATCH	191	W	39742	46011	6270	
		VMBATCH	192	W	46012	54371	8360	
		VMBATCH	193	W	54372	60641	6270	
		VMBATCH	194	W	60642	61059	418	
		VMBATCH1	191	W	61060	62731	1672	
		FSFCNTRL	191	MR	62732	69001	6270	
		FSFCNTRL	192	MR	69002	71091	2090	
		FSFCNTRL	193	MR	71092	73181	2090	
		FSFCNTRL	194	MR	73182	74435	1254	
		FSFCNTRL	195	MR	74436	75689	1254	
		FSFCNTRL	197	MR	75690	76107	418	
		FSFCNTRL	200	MR	76108	76525	418	
		FSFCNTRL	400	MR	76526	76943	418	
		FSFADMIN	192	MR	76944	77779	836	
		MAINT	324	MW	77780	90319	12540	
					90320	126015	35696	GAP

Figure 30 (Part 2 of 3). Directory Map Sample for a 3310 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSEXT	FB-512	MAINT	124	MW	000	126015	126016	
		SYSDUMP1	124	RR	000	126015	126016	
		$ALLOC$	B01	R	000	015	016	
		MAINT	296	MW	016	12555	12540	
		MAINT	319	MW	12556	55155	42600	
		MAINT	19E	MW	55156	80235	25080	
					80236	80247	012	GAP
		$TEMP$	B06	R	80248	100247	20000	
		MAINT	196	MW	100248	112787	12540	
					112788	115391	2604	GAP
		OPERATNS	193	MR	115392	122687	7296	
					122688	126015	3328	GAP
VMSRES	FB-512	DIRMAINT	123	MW	000	126015	126016	
		MAINT	123	MW	000	126015	126016	
		SYSDUMP1	123	RR	000	126015	126016	
		$ALLOC$	A01	R	000	015	016	
		$SAVSYS$	A02	R	016	9375	9360	
		MAINT	191	MW	9376	14657	5282	
		MAINT	194	MW	14658	35177	20520	
		OPERATOR	191	MR	35178	36201	1024	
		EREP	191	WR	36202	37113	912	
		OPERATNS	191	MR	37114	37569	456	
					37570	46911	9342	GAP
		MAINT	190	MW	46912	92479	45568	
		$DIRECT$	A04	R	92480	94303	1824	
		$SYSCKP$	A07	R	94304	94815	512	
		MAINT	19D	MW	94816	115715	20900	
		$TEMP$	A06	R	115716	122199	6484	
		$SYSWRM$	A07	R	122200	122711	512	
		$SYSERR$	A09	R	122712	123223	512	
		IVPM1	191	WR	123224	123479	256	
		IVPM2	191	WR	123480	123735	256	
		$CPNUC$	A10	R	123736	126015	2280	
VMSTGE	FB-512	MAINT	126	MW	000	126015	126016	
		SYSDUMP1	126	RR	000	126015	126016	
		$ALLOC$	D01	R	000	015	016	
					016	031	016	GAP
		MAINT	393	WR	032	68197	68166	
		MAINT	396	WR	68198	97457	29260	
					97458	126015	28558	GAP
VMSTG2	FB-512	$ALLOC$	D02	R	000	015	016	
					016	031	016	GAP
		MAINT	394	WR	032	71095	71064	
					71096	126015	54920	GAP

Figure 30 (Part 3 of 3). Directory Map Sample for a 3310 Device

3310 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3310    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
***********************************************************************
*
*       C L U S T E R     A N D     L I N E     C O N T R O L L E R
*
* THE ICA LINES ARE SHOWN AS COMMENTS.
*
***********************************************************************
         SPACE
*CLUST032 CLUSTER  CUTYPE=3274,GPOLL=407F,LINE=032
*        TERMINAL TERM=3278,SELECT=6040,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C1,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C2,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C3,MODEL=2
*CLUST033 CLUSTER  CUTYPE=3274,GPOLL=407F,LINE=033
*        TERMINAL TERM=3278,SELECT=6040,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C1,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C2,MODEL=2
*        TERMINAL TERM=3278,SELECT=60C3,MODEL=2
*
*      RDEVICE ADDRESS=(030,2),DEVTYPE=ICA,ADAPTER=BSCA
*      RDEVICE ADDRESS=032,DEVTYPE=ICA,ADAPTER=BSCA,CLUSTER=CLUST032
*      RDEVICE ADDRESS=033,DEVTYPE=ICA,ADAPTER=BSCA,CLUSTER=CLUST033
*      RDEVICE ADDRESS=034,DEVTYPE=2701,ADAPTER=TELE2
*      RDEVICE ADDRESS=(035,3),DEVTYPE=ICA,ADAPTER=BSCA
         EJECT
***********************************************************************
*                     C H A N N E L    Z E R O
*
*  NOTE: 009 DEFINED AS 3215 THAT CAN BE USED IN PRINTER/KEYBOARD MODE
*  NOTE: 010 AND 01F ARE DEFINED AS THE 3278 MODEL 2A SYSTEM CONSOLE
***********************************************************************
         SPACE
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
GRAF     RDEVICE ADDRESS=(012,12),DEVTYPE=3278,MODEL=2
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3278,MODEL=2A
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C,T)
         EJECT
```

Figure 31 (Part 1 of 3). Sample DMKRIO File for a 3310 Device

```
**********************************************************************
*
*                     C H A N N E L    O N E
*
**********************************************************************
         SPACE
GRAF     RDEVICE ADDRESS=(120,4),DEVTYPE=3277
GRAF     RDEVICE ADDRESS=(140,2),DEVTYPE=3278,MODEL=2
GRAF     RDEVICE ADDRESS=(142,2),DEVTYPE=3278,MODEL=3
GRAF     RDEVICE ADDRESS=(144,1),DEVTYPE=3278,MODEL=4
GRAF     RDEVICE ADDRESS=(145,1),DEVTYPE=3278,MODEL=5
GRAF     RDEVICE ADDRESS=(146,2),DEVTYPE=3279,MODEL=2
TAPE     RDEVICE ADDRESS=(180,4),DEVTYPE=3420,MODEL=4,FEATURE=DUALDENS
TAPE     RDEVICE ADDRESS=(1A0,2),DEVTYPE=3410,MODEL=3
         EJECT
**********************************************************************
*
*                     C H A N N E L    T W O
*
**********************************************************************
         SPACE
*  THE 3340 DEVICES ARE SHOWN AS COMMENTS
*
*DASD     RDEVICE ADDRESS=(200,4),DEVTYPE=3340
*DASD     RDEVICE ADDRESS=(210,4),DEVTYPE=3340
         SPACE
FBA3370  RDEVICE ADDRESS=(220,4),DEVTYPE=FB-512
FBA3370  RDEVICE ADDRESS=(230,4),DEVTYPE=FB-512
FBA3310  RDEVICE ADDRESS=(240,4),DEVTYPE=FB-512
FBA3310  RDEVICE ADDRESS=(250,4),DEVTYPE=FB-512
         EJECT
**********************************************************************
*
*                   C H A N N E L    T H R E E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(300,4),DEVTYPE=8809
         EJECT
**********************************************************************
*
*                     C O N T R O L     U N I T S
*
**********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=008,CUTYPE=3215
         RCTLUNIT ADDRESS=010,CUTYPE=3274
         RCTLUNIT ADDRESS=018,CUTYPE=3274
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=ICA
RCTLU1   RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3274,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=1A0,CUTYPE=3411
RCTLU2   RCTLUNIT ADDRESS=200,CUTYPE=FTA
         RCTLUNIT ADDRESS=210,CUTYPE=FTA
         RCTLUNIT ADDRESS=220,CUTYPE=FTA
         RCTLUNIT ADDRESS=230,CUTYPE=FTA
         RCTLUNIT ADDRESS=240,CUTYPE=FTA
         RCTLUNIT ADDRESS=250,CUTYPE=FTA
         RCTLUNIT ADDRESS=260,CUTYPE=FTA
         RCTLUNIT ADDRESS=270,CUTYPE=FTA
RCTLU3   RCTLUNIT ADDRESS=300,CUTYPE=FTA
         EJECT
```

Figure 31 (Part 2 of 3). Sample DMKRIO File for a 3310 Device

```
**********************************************************************
*
*                     C H A N N E L S
*
**********************************************************************
         SPACE
CHAN0    RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
CHAN1    RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
CHAN2    RCHANNEL ADDRESS=2,CHTYPE=FTA
CHAN3    RCHANNEL ADDRESS=3,CHTYPE=FTA
         EJECT
**********************************************************************
*
*                     C O N S O L E
*
**********************************************************************
         SPACE
         RIOGEN CONS=010,ALTCONS=(01F,018,009)
         END
```

Figure 31 (Part 3 of 3). Sample DMKRIO File for a 3310 Device

3310 DMKSYS File Supplied With The Product Tape

The 3310 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               VMSEXT,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=FB-512,                                         X
               SYSCLR=YES,                                             X
               SYSNUC=15467,                                           X
               SYSWRM=(15275,64),                                      X
               SYSERR=(15339,64),                                      X
               SYSCKP=(11788,64)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=2,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=2M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 32. Sample DMKSYS File for a 3310 Device

3310 DMKSNT Files Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT       VM/SP REL 3     FB-512 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSBLOK=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSBLOK - THE BLOCK    ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 33 (Part 1 of 8). Sample DMKSNT File for a 3310 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSBLOK=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
**********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
**********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
**********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 1119 (PAGE 139)
*      137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*     TOTAL = 138 PAGES
**********************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(2),                                            X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSBLOK=46912,                                          X
               VSYSRES=VMSRES
         EJECT
```

Figure 33 (Part 2 of 8). Sample DMKSNT File for a 3310 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 1120 (PAGE 140) TO BLK 2223 (PAGE 277)
*     TOTAL = 138 PAGES
********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(140),                                          X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSBLOK=46912,                                          X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2224 (PAGE 278) TO BLK 2359 (PAGE 294)
*     TOTAL = 17 PAGES
********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(278),                                          X
               SYSPGCT=16,                                             X
               SYSPGNM=(784-799),                                      X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               VSYSRES=,                                               X
               SYSBLOK=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2360 (PAGE 295) TO BLK 2751 (PAGE 343)
*     TOTAL = 49 PAGES
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(295),                                          X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 33 (Part 3 of 8). Sample DMKSNT File for a 3310 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2752 (PAGE 344) TO BLK 3655 (PAGE 456)
*     TOTAL = 113 PAGES
********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(344),                                          X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 3656 (PAGE 457) TO BLK 4815 (PAGE 601)
*     TOTAL = 145 PAGES
********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(457),                                          X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 4816 (PAGE 602) TO BLK 4951 (PAGE 618)
*     TOTAL = 17 PAGES
********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(602),                                          X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 33 (Part 4 of 8). Sample DMKSNT File for a 3310 Device

```
*********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 4952 (PAGE 619) TO BLK 5087 (PAGE 635)
*     TOTAL = 17 PAGES
**********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(619),                                          X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
**********************************************************************
*
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 5088 (PAGE 636) TO BLK 5223 (PAGE 652)
*     TOTAL = 17 PAGES
*
**********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(636),                                          X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
***********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 5224 (PAGE 653) TO BLK 5999 (PAGE 749)
*     TOTAL = 97 PAGES
**********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                      X
               SYSSIZE=384K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(653),                                          X
               SYSPGCT=96,                                             X
               SYSPGNM=(1072-1167),                                    X
               SYSHRSG=(67,68,69,70,71,72),                            X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 33 (Part 5 of 8). Sample DMKSNT File for a 3310 Device

```
*********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6000 (PAGE 750) TO BLK 6135 (PAGE 766)
*     TOTAL = 17 PAGES
*********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(750),                                          X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6136 (PAGE 767) TO BLK 6911 (PAGE 863)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(767),                                          X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
*********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6912 (PAGE 864) TO BLK 7687 (PAGE 960)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(864),                                          X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 33 (Part 6 of 8). Sample DMKSNT File for a 3310 Device

```
*********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 7688 (PAGE 961) TO BLK 8207 (PAGE 1025)
*     TOTAL = 65 PAGES
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(961),                                          X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 791 (PAGE 98)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(002),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 3120 (PAGE 390) TO BLK 3895 (PAGE 486)
*     TOTAL = 97 PAGES
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(390),SYSPGCT=96,                 X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 3896 (PAGE 487) TO BLK 4287 (PAGE 535)
*     TOTAL = 49 PAGES
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(487),                            X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 33 (Part 7 of 8). Sample DMKSNT File for a 3310 Device

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 4288 (PAGE 536) TO BLK 6215 (PAGE 776)
*     TOTAL = 241 PAGES
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(536),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 6216 (PAGE 777) TO BLK 9807 (PAGE 1225)
*     TOTAL = 449 PAGES
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(777),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 1559 (PAGE 194)
*     TOTAL = 193 PAGES
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(2),                                            X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 33 (Part 8 of 8). Sample DMKSNT File for a 3310 Device

VMSRES Pack Layout for 3310 Starter System

FB-512 Block Number	No. of FB-512 Blocks	Page Number	Number of Pages (BLK/8)	Contents
0-15	16	0-1	2	Allocation
16-9375	9360	2-1171	1170	Saved segments
9376-14657	5282	1172-	**	191 minidisk for MAINT
14658-35177	20520		**	194 minidisk for MAINT (CP)
35178- 36201	1024		**	191 minidisk for OPERATOR
36202-37113	912		**	191 minidisk for EREP
37114-37569	456		**	191 minidisk for OPERATNS
37570-46911	9342	-5863	**	xxx minidisk for user option
46912-92479	45568	5864-11559	5696	190 MAINT (CMS system mindisk)*
92480-94303	1824	11560-11787	228	VM/SP directory
94304-94815	512	11788-11851	64	Spool file checkpoint
94816-115715	20900	11852-	**	19D minidisk for MAINT (HELP)
115716-122199	6484	-15274	**	Spool space
122200-122711	512	15275-15338	64	Warm start data
122712-123223	512	15339-15402	64	I/O Error Recording area
123224-123479	256	15403-15434	32	191 minidisk for IVPM1
123480-123735	256	15435-15466	32	191 minidisk for IVPM2
123736-126015	2280	15467-15751	285	CP nucleus

Figure 34. VMSRES Layout for a 3310 Starter System

* The CMS nucleus must start on a 256 block boundary. 43520 displacement in the minidisk will contain the CMS nucleus. The CMSL nucleus must start on a 256 block boundary. 44544 displacement in the minidisk will contain the CMSL nucleus.

** Not on a page boundary.

Appendix K. 3330 Based System Sample Files

These directories are intended to be used as *samples only*. They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3330 Directory Supplied with the Product Tape

Sample Directory for a 3330 Device

```
***************************************************************
*  3330    SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 3330 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$  NOLOG
 MDISK A01 3330 000 001 VMSRES R
 MDISK B01 3330 000 001 VMSEXT R
 MDISK C01 3330 000 001 VMPK01 R
 MDISK C02 3330 000 001 VMPK02 R
 MDISK C03 3330 000 001 VMPK03 R
 MDISK D01 3330 000 001 VMSTGE R
 MDISK E01 3330 000 001 PROFPK R
 MDISK F01 3330 000 001 SQLPK  R
*
USER $TEMP$   NOLOG
 MDISK A03 3330 200 075 VMSRES R
 MDISK B03 3330 297 095 VMSEXT R
 MDISK C03 3330 215 120 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK B04 3330 001 110 VMSEXT R
 MDISK C04 3330 120 095 VMPK01 R
*
```

Figure 35 (Part 1 of 13). Sample Directory for a 3330 Device

```
USER $CPNUC$ NOLOG
 MDISK A09 3330 397 007 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A04 3330 275 004 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 3330 001 020 VMSRES R
 MDISK C02 3330 001 024 VMPK01 R
 MDISK A04 3330 001 004 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A05 3330 393 002 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A06 3330 198 002 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 3330 391 002 VMSRES R
*
***************************************************************
*             SYSTEM RELATED USERIDS
***************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 100 005 VMPK02 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 094 001 VMSRES MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 3330 211 002 VMPK02 MR RBATCH WBATCH MBATCH
*
```

Figure 35 (Part 2 of 13). Sample Directory for a 3330 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 228 005 VMPK02 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 3330 105 001 VMPK02 MR RCPRM WCPRM
 MDISK 192 3330 106 015 VMPK02 MR ALL WCPRM
 MDISK 291 3330 121 002 VMPK02 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 001 010 VMPK03 MR RCSP WCSP MCSP
 MDISK 193 3330 011 014 VMPK03 MR RCSP WCSP MCSP
 MDISK 502 3330 025 045 VMPK03 MR RCSP WCSP MCSP
 MDISK 503 3330 070 045 VMPK03 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 3330 115 005 VMPK01 M RMOVR WMOVR MMOVR
*
```

Figure 35 (Part 3 of 13). Sample Directory for a 3330 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3330 216 012 VMPK02 MR RDIRM WDIRM MDIRM
 MDISK 193 3330 170 017 VMPK02 MR RDIRM WDIRM MDIRM
 MDISK 195 3330 078 016 VMSRES MR RDIRM WDIRM MDIRM
 MDISK 123 3330 000 404 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 095 001 VMSRES MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 3330 281 002 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 3330 301 002 VMPK02 MR RFSF1 WFSF1
*
```

Figure 35 (Part 4 of 13). Sample Directory for a 3330 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 3330 303 015 VMPK02 MR RFSF WFSF
 MDISK 192 3330 318 005 VMPK02 MR RFSF WFSF
 MDISK 193 3330 323 005 VMPK02 MR RFSF WFSF
 MDISK 194 3330 328 003 VMPK02 MR RFSF WFSF
 MDISK 195 3330 331 003 VMPK02 MR RFSF WFSF
 MDISK 197 3330 334 001 VMPK02 MR RFSF WFSF
 MDISK 200 3330 335 001 VMPK02 MR RFSF WFSF
 MDISK 400 3330 336 001 VMPK02 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 001 027 VMPK02 MR ALL WIIPS MIIPS
 MDISK 193 3330 028 033 VMPK02 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 3330 402 001 VMSEXT MR RPSR WPSR MPSR
*
```

Figure 35 (Part 5 of 13). Sample Directory for a 3330 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 3330 096 004 VMPK02 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 158 002 VMPK02 MR RISPF WISPF MISPF
 MDISK 192 3330 061 035 VMPK02 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M   16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3330  395 001 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M   4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3330 396 001 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 35 (Part 6 of 13). Sample Directory for a 3330 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 3330 000 404 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 124 3330 000 404 VMSEXT MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3330 000 404 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3330 000 404 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 127 3330 000 404 VMPK02 MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3330 098 100 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 3330 021 012 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 3330 033 045 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 3330 283 030 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D 3330 331 060 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E 3330 135 060 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 201 3330 025 045 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 293 3330 335 024 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3330 359 023 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 3330 070 030 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 3330 105 010 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 310 3330 123 035 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 319 3330 195 090 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 31A 3330 397 005 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 324 3330 353 030 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 325 3330 187 020 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 390 3330 213 003 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 393 3330 001 150 VMSTGE WR MAINT   WMAINT
 MDISK 394 3330 234 170 VMSTGE WR RMAINT  WMAINT
 MDISK 396 3330 151 070 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 3330 382 015 VMPK01 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF 3330 000 808 CEPACK MR READ WRITE
*
```

Figure 35 (Part 7 of 13). Sample Directory for a 3330 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 096 002 VMSRES MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 313 001 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 3330 314 016 VMSRES MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 279 002 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT  250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3330 180 006 PROFPK MR RDBM WDBM MDBM
 MDISK 161 3330 186 020 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD 3330 254 024 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE 3330 230 024 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF 3330 206 024 PROFPK MR RDBM WDBM MDBM
*
```

Figure 35 (Part 8 of 13). Sample Directory for a 3330 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT  250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 3330 107 015 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 3330 122 006 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 397 006 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 3330 XXX 002 XXXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 191 3330 100 005 VMPK01 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT  250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3330 101 006 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 35 (Part 9 of 13). Sample Directory for a 3330 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3330 128 003 PROFPK MR RCAL WCAL MCAL
 MDISK 196 3330 131 018 PROFPK MR RCAL WCAL MCAL
*
************************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
************************************************************************
*
USER SMART  SMART  1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3330 337 016 VMPK02 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3330 001 022 SQLPK W
 MDISK 193 3330 023 051 SQLPK R  RSQL WSQL
 MDISK 195 3330 074 027 SQLPK RR RSQL WSQL MSQL
 MDISK 200 3330 101 064 SQLPK R  RSQL WSQL
 MDISK 201 3330 165 022 SQLPK R  RSQL WSQL
 MDISK 202 3330 187 212 SQLPK R  RSQL WSQL
*
USER SQLUSER SQLUSER  2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 3330 399 005 SQLPK W
*
```

Figure 35 (Part 10 of 13). Sample Directory for a 3330 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT  250 SYSADMIN
 IPL CMS
 CONSOLE009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 3330 005 010 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 3330 015 023 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 3330 077 024 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 3330 149 031 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 392 005 VMSEXT MR RDUMP WDUMP MDUMP
 MDISK 123 3330 000 404 VMSRES RR
 MDISK 124 3330 000 404 VMSEXT RR
 MDISK 125 3330 000 404 VMPK01 RR
 MDISK 126 3330 000 404 VMSTGE RR
 MDISK 127 3330 000 404 VMPK02 RR
 MDISK 128 3330 000 404 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 160 010 VMPK02 MR RVMAP WVMAP MVMAP
*
```

Figure 35 (Part 11 of 13). Sample Directory for a 3330 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3330 246 015 VMPK02 W RBATCH WBATCH
 MDISK 192 3330 261 020 VMPK02 W RBATCH WBATCH
 MDISK 193 3330 281 015 VMPK02 W RBATCH WBATCH
 MDISK 194 3330 296 001 VMPK02 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 3330 297 004 VMPK02 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3330 207 004 VMPK02 MR RUTIL WUTIL MUTIL
*
```

Figure 35 (Part 12 of 13). Sample Directory for a 3330 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3330 SYSTEM
*MDISK 150 3330 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3330 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3330 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3330 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3330 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3350 SYSTEM
*MDISK 350 3350 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3350 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3350 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3350 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 00000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 00000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 00000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 35 (Part 13 of 13). Sample Directory for a 3330 Device

Map of a 3330 Directory Sample

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	3330	OLTSEP	5FF	MR	000	807	808	
PROFPK	3330	SYSDUMP1	128	RR	000	403	404	
		$ALLOC$	E01	R	000	000	001	
		$SAVSYS$	A04	R	001	004	004	
		SYSADMIN	191	MR	005	014	010	
		SYSADMIN	298	MR	015	037	023	
					038	076	039	GAP
		SYSADMIN	398	MR	077	100	024	
		SFBATCH	191	MR	101	106	006	
		PROMAIL	191	MR	107	121	015	
		PROMAIL	151	MR	122	127	006	
		SFCAL	191	MR	128	130	003	
		SFCAL	196	MR	131	148	018	
		SYSADMIN	399	MR	149	179	031	
		PRODBM	191	MR	180	185	006	
		PRODBM	161	MR	186	205	020	
		PRODBM	5FF	MR	206	229	024	
		PRODBM	5FE	MR	230	253	024	
		PRODBM	5FD	MR	254	277	024	
					278	403	126	GAP
SQLPK	3330	$ALLOC$	F01	R	000	000	001	
		SQLDBA	191	W	001	022	022	
		SQLDBA	193	R	023	073	051	
		SQLDBA	195	RR	074	100	027	
		SQLDBA	200	R	101	164	064	
		SQLDBA	201	R	165	186	022	
		SQLDBA	202	R	187	398	212	
		SQLUSER	191	W	399	403	005	
VMPK01	3330	MAINT	125	MW	000	403	404	
		SYSDUMP1	125	RR	000	403	404	
		$ALLOC$	C01	R	000	000	001	
		$SAVSYS$	C02	R	001	024	024	
		MAINT	201	MW	025	069	045	
		MAINT	296	MW	070	099	030	
		RSCS	191	MR	100	104	005	
		MAINT	300	MW	105	114	010	
		DATAMOVR	191	M	115	119	005	
		$TDISK$	C04	R	120	214	095	
		$TEMP$	C03	R	215	334	120	
		MAINT	293	MW	335	358	024	
		MAINT	294	MW	359	381	023	
		MAINT	3A0	MW	382	396	015	
		PVM	191	MR	397	402	006	
					403	403	001	GAP

Figure 36 (Part 1 of 3). Directory Map Sample for a 3330 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK02	3330	MAINT	127	MW	000	403	404	
		SYSDUMP1	127	RR	000	403	404	
		$ALLOC$	C02	R	000	000	001	
		IIPS	191	MR	001	027	027	
		IIPS	193	MR	028	060	033	
		ISPVM	192	MR	061	095	035	
		ISMAINT	191	MR	096	099	004	
		ADMIN	191	MR	100	104	005	
		CPRM	191	MR	105	105	001	
		CPRM	192	MR	106	120	015	
		CPRM	291	MR	121	122	002	
		MAINT	310	MW	123	157	035	
		ISPVM	191	MR	158	159	002	
		VMAP	191	MR	160	169	010	
		DIRMAINT	193	MR	170	186	017	
		MAINT	325	MW	187	206	020	
		VMUTIL	191	MR	207	210	004	
		CMSBATCH	195	MR	211	212	002	
		MAINT	390	MW	213	215	003	
		DIRMAINT	191	MR	216	227	012	
		CMSUSER	191	MR	228	232	005	
					233	245	013	GAP
		VMBATCH	191	W	246	260	015	
		VMBATCH	192	W	261	280	020	
		VMBATCH	193	W	281	295	015	
		VMBATCH	194	W	296	296	001	
		VMBATCH1	191	W	297	300	004	
		FSFADMIN	192	MR	301	302	002	
		FSFCNTRL	191	MR	303	317	015	
		FSFCNTRL	192	MR	318	322	005	
		FSFCNTRL	193	MR	323	327	005	
		FSFCNTRL	194	MR	328	330	003	
		FSFCNTRL	195	MR	331	333	003	
		FSFCNTRL	197	MR	334	334	001	
		FSFCNTRL	200	MR	335	335	001	
		FSFCNTRL	400	MR	336	336	001	
		SMART	191	MR	337	352	016	
		MAINT	324	MW	353	382	030	
					383	403	021	GAP
VMPK03	3330	$ALLOC$	C03	R	000	000	001	
		CSPUSER	191	MR	001	010	010	
		CSPUSER	193	MR	011	024	014	
		CSPUSER	502	MR	025	069	045	
		CSPUSER	503	MR	070	114	045	
					115	403	289	GAP
VMSEXT	3330	MAINT	124	MW	000	403	404	
		SYSDUMP1	124	RR	000	403	404	
		$ALLOC$	B01	R	000	000	001	
		$TDISK$	B04	R	001	110	110	
					111	134	024	GAP
		MAINT	19E	MW	135	194	060	
		MAINT	319	MW	195	284	090	
					285	296	012	GAP
		$TEMP$	B03	R	297	391	095	
		SYSDUMP1	191	MR	392	396	005	
		MAINT	31A	MW	397	401	005	
		IPCS	191	MR	402	402	001	
					403	403	001	GAP

Figure 36 (Part 2 of 3). Directory Map Sample for a 3330 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSRES	3330	DIRMAINT	123	MW	000	403	404	
		MAINT	123	MW	000	403	404	
		SYSDUMP1	123	RR	000	403	404	
		$ALLOC$	A01	R	000	000	001	
		$SAVSYS$	A02	R	001	020	020	
		MAINT	191	MW	021	032	012	
		MAINT	194	MW	033	077	045	
		DIRMAINT	195	MR	078	093	016	
		AUTOLOG1	191	MR	094	094	001	
		DISKACNT	191	MR	095	095	001	
		OP1	191	MR	096	097	002	
		MAINT	190	MW	098	197	100	
		$SYSCKP$	A06	R	198	199	002	
		$TEMP$	A03	R	200	274	075	
		$DIRECT$	A04	R	275	278	004	
		OPERATOR	191	MR	279	280	002	
		EREP	191	WR	281	282	002	
		MAINT	196	MW	283	312	030	
		OPERATNS	191	MR	313	313	001	
		OPERATNS	193	MR	314	329	016	
					330	330	001	GAP
		MAINT	19D	MW	331	390	060	
		$SYSWRM$	A07	R	391	392	002	
		$SYSERR$	A05	R	393	394	002	
		IVPM1	191	WR	395	397	001	
		IVPM2	191	WR	396	398	001	
		$CPNUC$	A09	R	397	403	007	
VMSTGE	3330	MAINT	126	MW	000	403	404	
		SYSDUMP1	126	RR	000	403	404	
		$ALLOC$	D01	RR	000	000	001	
		MAINT	393	WR	001	150	150	
		MAINT	396	WR	151	220	070	
					221	233	013	GAP
		MAINT	394	WR	234	403	170	

Figure 36 (Part 3 of 3). Directory Map Sample for a 3330 Device

3330 DMKRIO File Supplied With Product Tape

```
RIO      TITLE 'DMKRIO - 3330    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
         COPY  OPTIONS
**********************************************************************
*
*       C L U S T E R   A N D   L I N E   C O N T R O L L E R S
*
**********************************************************************
         SPACE
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
       SPACE
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 37 (Part 1 of 4). Sample DMKRIO File for a 3330 Device

```
**********************************************************************
*
*                     C H A N N E L    Z E R O
*
* NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4331
*            OR 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*           016 MAY BE USED AS DISPLAY CONSOLE ON 168 AND 165-II.
*
**********************************************************************
         SPACE
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=A
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
GRAF     RDEVICE ADDRESS=(018,2),DEVTYPE=3278,MODEL=2
CONS     RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(030,16),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
**********************************************************************
*
*                     C H A N N E L    O N E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,32),DEVTYPE=3277
         RDEVICE ADDRESS=(140,16),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(150,16),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(160,16),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(170,16),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1C0,16),DEVTYPE=3330,MODEL=11
         EJECT
```

Figure 37 (Part 2 of 4). Sample DMKRIO File for a 3330 Device

```
************************************************************************
*
*                     C H A N N E L   T W O
*
************************************************************************
         SPACE
         RDEVICE ADDRESS=(220,32),DEVTYPE=3277
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3350
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         EJECT
************************************************************************
*
*                     C H A N N E L   T H R E E
*
************************************************************************
         SPACE
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3350
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         EJECT
************************************************************************
*
*                     C H A N N E L   F O U R
*
************************************************************************
         SPACE
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3350
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
************************************************************************
*
*                     C H A N N E L   F I V E
*
************************************************************************
         SPACE
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
         EJECT
```

Figure 37 (Part 3 of 4). Sample DMKRIO File for a 3330 Device

```
***********************************************************************
*
*                     C O N T R O L    U N I T S
*
***********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=32-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=160,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU2   RCTLUNIT ADDRESS=220,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803
         RCTLUNIT ADDRESS=580,CUTYPE=3803
         EJECT
***********************************************************************
*
*                       C H A N N E L S
*
***********************************************************************
         SPACE
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
***********************************************************************
*
*                C O N S O L E    D E F I N I T I O N
*
***********************************************************************
         SPACE
         RIOGEN  CONS=014,ALTCONS=(01F,009,01A,016,010)
         END
```

Figure 37 (Part 4 of 4). Sample DMKRIO File for a 3330 Device

3330 DMKSYS File Supplied With The Product Tape

The 3330 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               VMSEXT,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=3330,                                           X
               SYSCLR=YES,                                             X
               SYSNUC=397,                                             X
               SYSWRM=(391,2),                                         X
               SYSERR=(393,2),                                         X
               SYSCKP=(198,2)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 38. Sample DMKSYS File for a 3330 Device

3330 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT       VM/SP REL 3       3330 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSCYL=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSCYL  - THE CYLINDER ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 39 (Part 1 of 8). Sample DMKSNT File for a 3330 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSCYL=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
*********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
*********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*     ( THE ALLOCATIONS ARE BASED ON 57 PAGES/3330 CYLINDER )
*          CYL 1, PAGE 1 TO CYL 3, PAGE 24 (138 PAGES)
*         137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*********************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(001,1),                                        X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=98,                                              X
               VSYSRES=VMSRES
         EJECT
```

Figure 39 (Part 2 of 8). Sample DMKSNT File for a 3330 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 25 TO CYL 5, PAGE 48 (138 PAGES)
*********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(003,25),                                       X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=98,                                              X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 5, PAGE 49 TO CYL 6, PAGE 08 ( 17 PAGES )
*********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(005,49),                                       X
               SYSPGNM=(784-799),                                      X
               SYSPGCT=16,                                             X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 6, PAGE 09 TO CYL 6, PAGE 57 (49 PAGES)
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(006,09),                                       X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 39 (Part 3 of 8). Sample DMKSNT File for a 3330 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 7, PAGE 01 TO CYL 8, PAGE 56 ( 113 PAGES )
*********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(007,01),                                       X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 8, PAGE 57 TO CYL 11, PAGE 30 (145 PAGES)
*********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(008,57),                                       X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 11, PAGE 31 TO CYL 11, PAGE 47 (17 PAGES)
********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(011,31),                                       X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 39 (Part 4 of 8). Sample DMKSNT File for a 3330 Device

```
*********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 11, PAGE 48 TO CYL 12, PAGE 07 (17 PAGES)
*********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                         X
               CPSIZE=48K,                                            X
               CPTYPE=EP,                                             X
               SYSSTRT=(011,048),                                     X
               SYSPGCT=16,                                            X
               SYSVOL=VMSRES
         EJECT
*********************************************************************
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 12, PAGE 08 TO CYL 12, PAGE 24 (17 PAGES)
*********************************************************************
VMEP02 NAMENCP CPNAME=VMEP02,                                         X
               CPSIZE=48K,                                            X
               CPTYPE=EP,                                             X
               SYSSTRT=(012,008),                                     X
               SYSPGCT=16,                                            X
               SYSVOL=VMSRES
         EJECT
**********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 12, PAGE 25 TO CYL 14, PAGE 07 (97 PAGES)
*********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                     X
               SYSSIZE=384K,                                          X
               SYSVOL=VMSRES,                                         X
               SYSCYL=,                                               X
               SYSSTRT=(012,025),                                     X
               SYSPGCT=96,                                            X
               SYSPGNM=(1072-1167),                                   X
               SYSHRSG=(67,68,69,70,71,72),                           X
               VSYSRES=,                                              X
               VSYSADR=IGNORE
         EJECT
```

Figure 39 (Part 5 of 8). Sample DMKSNT File for a 3330 Device

```
*********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 14, PAGE 08 TO CYL 14, PAGE 24 (17 PAGES)
*********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(014,008),                                      X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 14, PAGE 25 TO CYL 16, PAGE 07 (97 PAGES)
*********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(014,25),                                       X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
*********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 16, PAGE 08 TO CYL 17, PAGE 47 (97 PAGES)
*********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(016,08),                                       X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 39 (Part 6 of 8). Sample DMKSNT File for a 3330 Device

```
*********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 17, PAGE 48 TO CYL 18, PAGE 55 (65 PAGES)
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(017,48),                                       X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE 40 (97 PAGES)
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(001,01),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 7, PAGE 47 TO CYL 9, PAGE 29 (97 PAGES)
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(007,47),SYSPGCT=96,              X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 09, PAGE 30 TO CYL 10, PAGE 21 (49 PAGES)
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(009,030),                        X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 39 (Part 7 of 8). Sample DMKSNT File for a 3330 Device

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 10, PAGE 22 TO CYL 14, PAGE 34 (241 PAGES)
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(10,22),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 14, PAGE 35 TO CYL 22, PAGE 27 (449 PAGES)
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(014,35),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*    CYL 1 (PAGE 01) TO CYL 4 (PAGE 022)
*    TOTAL 193 PAGES
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 39 (Part 8 of 8). Sample DMKSNT File for a 3330 Device

VMSRES Pack Layout for 3330 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-20	20	Reserved for saved segments
21-32	12	191 minidisk for MAINT
33-77	45	194 minidisk for MAINT (CP)
78-97	20	xxx minidisk for user option
98-197	100	190 minidisk for MAINT (CMS system disk)*
198-199	2	Spool file checkpoint
200-274	75	Page/spool space
275-278	4	VM/SP directory
279-280	2	191 minidisk for OPERATOR
281-282	2	191 minidisk for EREP
283-299	17	xxx minidisk for user option
300	1	191 minidisk for OPERATNS
301-316	16	193 minidisk for OPERATNS
317-330	14	xxx minidisk for user option
331-390	60	19D minidisk for MAINT (HELP)
391-392	2	Warm start data
393-394	2	I/O Error Recording area
395	1	191 minidisk for IVPM1 user
396	1	191 minidisk for IVPM2 user
397-403	7	CP nucleus

Figure 40. VMSRES Layout for a 3330 Starter System

* Specify cylinder 94 to load the CMS nucleus which occupies the first 3 cylinders of the recompiled area.
Specify cylinder 97 to load the CMSL nucleus which occupies the last 3 cylinders of the recompiled area.

Appendix L. 3340 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3340 Directory Supplied with the Product Tape

Sample Directory for a 3340 Device

```
***************************************************************
*  3340    SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 3340 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$  NOLOG
 MDISK A01 3340 000 001 VMSRES R
 MDISK B01 3340 000 001 VMSEXT R
 MDISK C01 3340 000 001 VMPK01 R
 MDISK C02 3340 000 001 VMPK02 R
 MDISK C03 3340 000 001 VMPK03 R
 MDISK D01 3340 000 001 VMSTGE R
 MDISK D02 3340 000 001 VMSTG2 R
 MDISK E01 3340 000 001 PROFPK R
 MDISK F01 3340 000 001 SQLPK  R
 MDISK F02 3340 000 001 SQLPK2 R
*
USER $TEMP$ NOLOG
 MDISK A03 3340 095 030 VMSRES R
 MDISK B08 3340 465 161 VMSEXT R
 MDISK C03 3340 286 081 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK B02 3340 367 250 VMPK01 R
*
```

Figure 41 (Part 1 of 13). Sample Directory for a 3340 Device

```
USER $CPNUC$ NOLOG
 MDISK A09 3340 032 012 VMSRES R
*
USER $UNUSED$ NOLOG
 MDISK A09 3340 029 003 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A04 3340 070 010 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 3340 001 028 VMSRES R
 MDISK B02 3340 001 065 VMPK01 R
 MDISK A04 3340 001 009 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A05 3340 080 005 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A06 3340 090 005 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 3340 085 005 VMSRES R
*
***************************************************************
*              SYSTEM RELATED USERIDS
***************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 387 012 VMPK02 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 692 002 VMPK01 MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 3340 691 005 VMSEXT MR RBATCH WBATCH MBATCH
*
```

Figure 41 (Part 2 of 13). Sample Directory for a 3340 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 194 012 VMPK01 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 3340 341 003 VMPK02 MR RCPRM WCPRM
 MDISK 192 3340 344 038 VMPK02 MR ALL WCPRM
 MDISK 291 3340 382 005 VMPK02 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 348 023 VMPK03 MR RCSP WCSP MCSP
 MDISK 193 3340 371 035 VMPK03 MR RCSP WCSP MCSP
 MDISK 502 3340 406 113 VMPK03 MR RCSP WCSP MCSP
 MDISK 503 3340 519 113 VMPK03 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 3340 678 012 VMPK01 M RMOVR WMOVR MMOVR
*
```

Figure 41 (Part 3 of 13). Sample Directory for a 3340 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3340 582 024 VMPK02 MR RDIRM WDIRM MDIRM
 MDISK 193 3340 487 041 VMPK02 MR RDIRM WDIRM MDIRM
 MDISK 195 3340 442 041 VMSRES MR RDIRM WDIRM MDIRM
 MDISK 123 3340 000 696 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 690 002 VMPK01 MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 3340 363 004 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 3340 303 005 VMPK03 MR RFSF1 WFSF1
*
```

Figure 41 (Part 4 of 13). Sample Directory for a 3340 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 3340 214 038 VMPK03 MR RFSF WFSF
 MDISK 192 3340 252 013 VMPK03 MR RFSF WFSF
 MDISK 193 3340 265 013 VMPK03 MR RFSF WFSF
 MDISK 194 3340 278 008 VMPK03 MR RFSF WFSF
 MDISK 195 3340 286 008 VMPK03 MR RFSF WFSF
 MDISK 197 3340 294 003 VMPK03 MR RFSF WFSF
 MDISK 200 3340 297 003 VMPK03 MR RFSF WFSF
 MDISK 400 3340 300 003 VMPK03 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 001 065 VMPK02 MR ALL WIIPS MIIPS
 MDISK 193 3340 066 090 VMPK02 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 3340 156 002 VMPK02 MR RPSR WPSR MPSR
*
```

Figure 41 (Part 5 of 13). Sample Directory for a 3340 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 3340 168 012 VMPK02 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 158 010 VMPK02 MR RISPF WISPF MISPF
 MDISK 192 3340 399 088 VMPK02 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M   16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3340  214 002 VMPK01 WR READ WRITE
*
USER IVPM2 IVPASS 3M   4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3340 216 002 VMPK01 WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 41 (Part 6 of 13). Sample Directory for a 3340 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 3340 000 696 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 124 3340 000 696 VMSEXT MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3340 000 696 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3340 000 696 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 127 3340 000 696 VMPK02 MW RSYSRES WSYSRES MSYSRES
 MDISK 128 3340 000 696 VMPK03 MW RSYSRES WSYSRES MSYSRES
 MDISK 129 3340 000 696 VMSTG2 MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3340 125 236 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 3340 044 026 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 3340 589 107 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 3340 367 075 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D 3340 489 100 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E 3340 312 150 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 201 3340 066 107 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 293 3340 617 061 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3340 218 058 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 3340 012 075 VMSEXT MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 3340 268 063 VMPK02 MW RMAINT  WMAINT MMAINT
 MDISK 310 3340 180 088 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 319 3340 087 225 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 31A 3340 679 012 VMSEXT MW ALL     WMAINT MMAINT
 MDISK 324 3340 606 075 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 325 3340 528 048 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 390 3340 579 003 VMPK02 MW ALL     WMAINT MMAINT
 MDISK 393 3340 001 350 VMSTG2 WR RMAINT  WMAINT
 MDISK 394 3340 301 395 VMSTGE WR RMAINT  WMAINT
 MDISK 396 3340 001 175 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 3340 037 038 VMPK03 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF 3340 000 696 CEPACK MR READ WRITE
*
```

Figure 41 (Part 7 of 13). Sample Directory for a 3340 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 576 003 VMPK02 MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 626 001 VMSEXT MR RIPCS WIPCS MIPCS
 MDISK 193 3340 627 040 VMSEXT MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 361 002 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT 250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3340 330 015 PROFPK MR RDBM WDBM MDBM
 MDISK 161 3340 345 050 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD 3340 530 060 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE 3340 470 060 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF 3340 410 060 PROFPK MR RDBM WDBM MDBM
*
```

Figure 41 (Part 8 of 13). Sample Directory for a 3340 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT 250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 3340 635 038 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 3340 395 015 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 206 008 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 3340 XXX 002 XXXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 191 3340 001 012 VMPK03 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3340 681 015 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 41 (Part 9 of 13). Sample Directory for a 3340 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3340 673 008 PROFPK MR RCAL WCAL MCAL
 MDISK 196 3340 590 045 PROFPK MR RCAL WCAL MCAL
*
************************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
************************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3340 308 040 VMPK03 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3340 001 057 SQLPK W
 MDISK 193 3340 058 132 SQLPK R  RSQL WSQL
 MDISK 195 3340 190 069 SQLPK RR RSQL WSQL MSQL
 MDISK 200 3340 259 156 SQLPK R  RSQL WSQL
 MDISK 201 3340 415 050 SQLPK R  RSQL WSQL
 MDISK 202 3340 001 480 SQLPK2 R  RSQL WSQL
*
USER SQLUSER SQLUSER  2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 3340 465 012 SQLPK W
*
```

Figure 41 (Part 10 of 13). Sample Directory for a 3340 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT  250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 3340 010 023 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 3340 033 054 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 3340 200 057 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 3340 257 073 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 667 012 VMSEXT MR RDUMP WDUMP MDUMP
 MDISK 123 3340 000 696 VMSRES RR
 MDISK 124 3340 000 696 VMSEXT RR
 MDISK 125 3340 000 696 VMPK01 RR
 MDISK 126 3340 000 696 VMSTGE RR
 MDISK 127 3340 000 696 VMPK02 RR
 MDISK 128 3340 000 696 VMPK03 RR
 MDISK 129 3340 000 696 VMSTG2 RR
 MDISK 130 3340 000 696 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 013 024 VMPK03 MR RVMAP WVMAP MVMAP
*
```

Figure 41 (Part 11 of 13). Sample Directory for a 3340 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3340 075 038 VMPK03 W RBATCH WBATCH
 MDISK 192 3340 113 050 VMPK03 W RBATCH WBATCH
 MDISK 193 3340 163 038 VMPK03 W RBATCH WBATCH
 MDISK 194 3340 201 003 VMPK03 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 3340 204 010 VMPK03 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3340 331 009 VMPK02 MR RUTIL WUTIL MUTIL
*
```

Figure 41 (Part 12 of 13). Sample Directory for a 3340 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3330 SYSTEM
*MDISK 150 3330 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3330 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3330 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3330 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3330 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3350 SYSTEM
*MDISK 350 3350 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3350 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3350 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3350 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 00000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 00000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 00000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 41 (Part 13 of 13). Sample Directory for a 3340 Device

Map of a 3340 Directory Sample

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	3340	OLTSEP	5FF	MR	000	695	696	
PROFPK	3340	SYSDUMP1	130	RR	000	695	696	
		$ALLOC$	E01	R	000	000	001	
		$SAVSYS$	A04	R	001	009	009	
		SYSADMIN	191	MR	010	032	023	
		SYSADMIN	298	MR	033	086	054	
					087	199	113	GAP
		SYSADMIN	398	MR	200	256	057	
		SYSADMIN	399	MR	257	329	073	
		PRODBM	191	MR	330	344	015	
		PRODBM	161	MR	345	394	050	
		PROMAIL	151	MR	395	409	015	
		PRODBM	5FF	MR	410	469	060	
		PRODBM	5FE	MR	470	529	060	
		PRODBM	5FD	MR	530	589	060	
		SFCAL	196	MR	590	634	045	
		PROMAIL	191	MR	635	672	038	
		SFCAL	191	MR	673	680	008	
		SFBATCH	191	MR	681	695	015	
SQLPK	3340	$ALLOC$	F01	R	000	000	001	
		SQLDBA	191	W	001	057	057	
		SQLDBA	193	R	058	189	132	
		SQLDBA	195	RR	190	258	069	
		SQLDBA	200	R	259	414	156	
		SQLDBA	201	R	415	464	050	
		SQLUSER	191	W	465	476	012	
					477	695	219	GAP
SQLPK2	3340	$ALLOC$	F02	R	000	000	001	
		SQLDBA	202	R	001	480	480	
					481	695	215	GAP
VMPK01	3340	MAINT	125	MW	000	695	696	
		SYSDUMP1	125	RR	000	695	696	
		$ALLOC$	C01	R	000	000	001	
		$SAVSYS$	B02	R	001	065	065	
		MAINT	201	MW	066	172	107	
					173	193	021	GAP
		CMSUSER	191	MR	194	205	012	
		PVM	191	MR	206	213	008	
		IVPM1	191	WR	214	215	002	
		IVPM2	191	WR	216	217	002	
		MAINT	294	MW	218	275	058	
					276	285	010	GAP
		$TEMP$	C03	R	286	366	081	
		$TDISK$	B02	R	367	616	250	
		MAINT	293	MW	617	677	061	
		DATAMOVR	191	M	678	689	012	
		DISKACNT	191	MR	690	691	002	
		AUTOLOG1	191	MR	692	693	002	
					694	695	002	GAP

Figure 42 (Part 1 of 3). Directory Map Sample for a 3340 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK02	3340	MAINT	127	MW	000	695	696	
		SYSDUMP1	127	RR	000	695	696	
		$ALLOC$	C02	R	000	000	001	
		IIPS	191	MR	001	065	065	
		IIPS	193	MR	066	155	090	
		IPCS	191	MR	156	157	002	
		ISPVM	191	MR	158	167	010	
		ISMAINT	191	MR	168	179	012	
		MAINT	310	MW	180	267	088	
		MAINT	300	MW	268	330	063	
		VMUTIL	191	MR	331	339	009	
					340	340	001	GAP
		CPRM	191	MR	341	343	003	
		CPRM	192	MR	344	381	038	
		CPRM	291	MR	382	386	005	
		ADMIN	191	MR	387	398	012	
		ISPVM	192	MR	399	486	088	
		DIRMAINT	193	MR	487	527	041	
		MAINT	325	MW	528	575	048	
		OP1	191	MR	576	578	003	
		MAINT	390	MW	579	581	003	
		DIRMAINT	191	MR	582	605	024	
		MAINT	324	MW	606	680	075	
					681	695	015	GAP
VMPK03	3340	MAINT	128	MW	000	695	696	
		SYSDUMP1	128	RR	000	695	696	
		$ALLOC$	C03	R	000	000	001	
		RSCS	191	MR	001	012	012	
		VMAP	191	MR	013	036	024	
		MAINT	3A0	MW	037	074	038	
		VMBATCH	191	W	075	112	038	
		VMBATCH	192	W	113	162	050	
		VMBATCH	193	W	163	200	038	
		VMBATCH	194	W	201	203	003	
		VMBATCH1	191	W	204	213	010	
		FSFCNTRL	191	MR	214	251	038	
		FSFCNTRL	192	MR	252	264	013	
		FSFCNTRL	193	MR	265	277	013	
		FSFCNTRL	194	MR	278	285	008	
		FSFCNTRL	195	MR	286	293	008	
		FSFCNTRL	197	MR	294	296	003	
		FSFCNTRL	200	MR	297	299	003	
		FSFCNTRL	400	MR	300	302	003	
		FSFADMIN	192	MR	303	307	005	
		SMART	191	MR	308	347	040	
		CSPUSER	191	MR	348	370	023	
		CSPUSER	193	MR	371	405	035	
		CSPUSER	502	MR	406	518	113	
		CSPUSER	503	MR	519	631	113	
					632	695	064	GAP

Figure 42 (Part 2 of 3). Directory Map Sample for a 3340 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSEXT	3340	MAINT	124	MW	000	695	696	
		SYSDUMP1	124	RR	000	695	696	
		$ALLOC$	B01	R	000	000	001	
					001	011	011	GAP
		MAINT	296	MW	012	086	075	
		MAINT	319	MW	087	311	225	
		MAINT	19E	MW	312	461	150	
					462	464	003	GAP
		$TEMP$	B08	R	465	625	161	
		OPERATNS	191	MR	626	626	001	
		OPERATNS	193	MR	627	666	040	
		SYSDUMP1	191	MR	667	678	012	
		MAINT	31A	MW	679	690	012	
		CMSBATCH	195	MR	691	695	005	
VMSRES	3340	DIRMAINT	123	MW	000	695	696	
		MAINT	123	MW	000	695	696	
		SYSDUMP1	123	RR	000	695	696	
		$ALLOC$	A01	R	000	000	001	
		$SAVSYS$	A02	R	001	028	028	
		$UNUSED$	A09	R	029	031	003	
		$CPNUC$	A09	R	032	043	012	
		MAINT	191	MW	044	069	026	
		$DIRECT$	A04	R	070	079	010	
		$SYSERR$	A05	R	080	084	005	
		$SYSWRM$	A07	R	085	089	005	
		$SYSCKP$	A06	R	090	094	005	
		$TEMP$	A03	R	095	124	030	
		MAINT	190	MW	125	360	236	
		OPERATOR	191	MR	361	362	002	
		EREP	191	WR	363	366	004	
		MAINT	196	MW	367	441	075	
		DIRMAINT	195	MR	442	482	041	
					483	488	006	GAP
		MAINT	19D	MW	489	588	100	
		MAINT	194	MW	589	695	107	
VMSTGE	3340	MAINT	126	MW	000	695	696	
		SYSDUMP1	126	RR	000	695	696	
		$ALLOC$	D01	R	000	000	001	
		MAINT	396	WR	001	175	175	
					176	300	125	GAP
		MAINT	394	WR	301	695	395	
VMSTG2	3340	MAINT	129	MW	000	695	696	
		SYSDUMP1	129	RR	000	695	696	
		$ALLOC$	D02	R	000	000	001	
		MAINT	393	WR	001	350	350	
					351	695	345	GAP

Figure 42 (Part 3 of 3). Directory Map Sample for a 3340 Device

3340 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3340    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
         COPY  OPTIONS
**********************************************************************
*
*       C L U S T E R S   A N D    L I N E    C O N T R O L L E R S
*
**********************************************************************
         SPACE
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
       SPACE
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 43 (Part 1 of 4). Sample DMKRIO File for a 3340 Device

```
***********************************************************************
*
*                     C H A N N E L    Z E R O
*
* NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4331
*            OR 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*           016 MAY BE USED AS DISPLAY CONSOLE ON 168 AND 165-II.
*
***********************************************************************
         SPACE
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=A
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
GRAF     RDEVICE ADDRESS=(018,2),DEVTYPE=3278,MODEL=2
CONS     RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(030,16),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
***********************************************************************
*
*                     C H A N N E L    O N E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,32),DEVTYPE=3277
         RDEVICE ADDRESS=(140,16),DEVTYPE=3340
         RDEVICE ADDRESS=(150,16),DEVTYPE=3340
         RDEVICE ADDRESS=(160,16),DEVTYPE=3340
         RDEVICE ADDRESS=(170,16),DEVTYPE=3340
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1C0,64),DEVTYPE=3340
         EJECT
```

Figure 43 (Part 2 of 4). Sample DMKRIO File for a 3340 Device

```
***********************************************************************
*
*                     C H A N N E L    T W O
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(200,8),DEVTYPE=3340
         RDEVICE ADDRESS=(210,8),DEVTYPE=3340
         RDEVICE ADDRESS=(220,32),DEVTYPE=3277
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3350
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         EJECT
***********************************************************************
*
*                     C H A N N E L    T H R E E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(300,4),DEVTYPE=8809
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3340
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         EJECT
***********************************************************************
*
*                     C H A N N E L    F O U R
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3350
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
***********************************************************************
*
*                     C H A N N E L    F I V E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
         EJECT
```

Figure 43 (Part 3 of 4). Sample DMKRIO File for a 3340 Device

```
**********************************************************************
*
*                     C O N T R O L   U N I T S
*
**********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=32-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3830,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1C0,CUTYPE=3830,FEATURE=64-DEVICE
RCTLU2   RCTLUNIT ADDRESS=200,CUTYPE=3830,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=210,CUTYPE=3830,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=220,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=300,CUTYPE=FTA
         RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803
         RCTLUNIT ADDRESS=580,CUTYPE=3803
         EJECT
**********************************************************************
*
*                         C H A N N E L S
*
**********************************************************************
         SPACE
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
**********************************************************************
*
*                        C O N S O L E
*
**********************************************************************
         SPACE
         RIOGEN  CONS=014,ALTCONS=(01F,009,01A,016,010)
         END
```

Figure 43 (Part 4 of 4). Sample DMKRIO File for a 3340 Device

3340 DMKSYS File Supplied With The Product Tape

The 3340 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               VMSEXT,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=3340,                                           X
               SYSCLR=YES,                                             X
               SYSNUC=32,                                              X
               SYSWRM=(85,5),                                          X
               SYSERR=(80,5),                                          X
               SYSCKP=(90,5)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 44. Sample DMKSYS File for a 3340 Device

3340 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT        VM/SP REL 3       3340 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSCYL=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSCYL  - THE CYLINDER ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 45 (Part 1 of 9). Sample DMKSNT File for a 3340 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSCYL=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
*********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
*********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*     ( THE ALLOCATIONS ARE BASED ON 24 PAGES/3340 CYLINDER )
*          CYL 1, PAGE 1 TO CYL 6, PAGE 18 (138 PAGES)
*         137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*********************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(001,1),                                        X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=125,                                             X
               VSYSRES=VMSRES
         EJECT
```

Figure 45 (Part 2 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 6, PAGE 19 TO CYL 12, PAGE 12 (138 PAGES)
**********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                      X
               SYSVOL=VMSRES,                                           X
               SYSSTRT=(006,19),                                        X
               SYSPGNM=(0-4,14-33,3840-3951),                           X
               SYSPGCT=137,                                             X
               SYSHRSG=(240,241,242,243,244,245,246),                   X
               SYSSIZE=256K,                                            X
               VSYSADR=190,                                             X
               SYSCYL=125,                                              X
               VSYSRES=VMSRES
         EJECT
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 12, PAGE 13 TO CYL 13, PAGE 05 (17 PAGES)
**********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                        X
               SYSVOL=VMSRES,                                           X
               SYSSTRT=(012,13),                                        X
               SYSPGNM=(784-799),                                       X
               SYSPGCT=16,                                              X
               SYSHRSG=(49),                                            X
               SYSSIZE=64K,                                             X
               SYSCYL=,                                                 X
               VSYSRES=,                                                X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 13, PAGE 06 TO CYL 15, PAGE 06 (49 PAGES)
*********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                        X
               SYSVOL=VMSRES,                                           X
               SYSSTRT=(013,06),                                        X
               SYSPGCT=48,                                              X
               SYSPGNM=(800-847),                                       X
               SYSHRSG=(50,51,52),                                      X
               SYSSIZE=192K,                                            X
               SYSCYL=,                                                 X
               VSYSRES=,                                                X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 45 (Part 3 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 15, PAGE 7 TO CYL 19, PAGE 23 (113 PAGES)
**********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(015,07),                                       X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 19, PAGE 24 TO CYL 25, PAGE 24 (145 PAGES)
**********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(019,24),                                       X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 26, PAGE 01 TO CYL 26, PAGE 17 (17 PAGES)
*********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(026,01),                                       X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 45 (Part 4 of 9). Sample DMKSNT File for a 3340 Device

```
********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 26, PAGE 18 TO CYL 27, PAGE 10 (17 PAGES)
********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(026,018),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
********************************************************************
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 27, PAGE 11 TO CYL 28, PAGE 03 (17 PAGES)
********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(027,011),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
```

Figure 45 (Part 5 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL  3, PAGE 18 TO CYL  4, PAGE 11 (17 PAGES)
*********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMPK01,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(003,018),                                      X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL  4, PAGE 12 TO CYL 08, PAGE 12 (97 PAGES)
*********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMPK01,                                          X
               SYSSTRT=(004,12),                                       X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
*********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL  8, PAGE 13 TO CYL 12, PAGE 13 (97 PAGES)
*********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMPK01,                                          X
               SYSSTRT=(008,13),                                       X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 45 (Part 6 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMPK01, AS FOLLOWS:
*          CYL 12, PAGE 14 TO CYL 15, PAGE 06 (65 PAGES)
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMPK01,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(012,14),                                       X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 15, PAGE 7 TO CYL 19, PAGE 07 (97 PAGES)
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(015,07),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 19, PAGE 8 TO CYL 23, PAGE 08 (97 PAGES)
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(019,08),SYSPGCT=96,              X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 23, PAGE 9 TO CYL 25, PAGE 09 (49 PAGES)
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(023,009),                        X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 45 (Part 7 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 25, PAGE 10 TO CYL 35, PAGE 10 (241 PAGES)
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(25,10),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 35, PAGE 11 TO CYL 54, PAGE 03 (449 PAGES)
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(035,11),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 54, PAGE 04 TO CYL 58, PAGE 04 (97 PAGES)
*********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                      X
               SYSSIZE=384K,                                           X
               SYSVOL=VMPK01,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(054,004),                                      X
               SYSPGCT=96,                                             X
               SYSPGNM=(1072-1167),                                    X
               SYSHRSG=(67,68,69,70,71,72),                            X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 45 (Part 8 of 9). Sample DMKSNT File for a 3340 Device

```
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 01, PAGE 01 TO CYL 09, PAGE 01 (193 PAGES)
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 45 (Part 9 of 9). Sample DMKSNT File for a 3340 Device

VMSRES Pack Layout for 3340-70 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-28	28	Reserved for saved segments
29-31	3	not used
32-43	12	CP nucleus
44-69	26	191 minidisk for MAINT
70-79	10	VM/SP directory
80-84	5	I/O Error Recording area
85-89	5	Warm start data
90-94	5	Spool file checkpoint
95-124	30	Page/spool space
125-360	236	190 minidisk for MAINT (CMS system disk)*
361-362	2	191 minidisk for OPERATOR
363-366	4	191 minidisk for EREP
367-488	122	xxx minidisk for user option
489-588	100	19D minidisk for MAINT (HELP)
589-695	107	194 minidisk for MAINT (CP)

Figure 46. VMSRES Layout for a 3340-70 Starter System

* Specify cylinder 222 to load the CMS nucleus which occupies the first 7 cylinders of the recompiled area.
Specify cylinder 229 to load the CMSL nucleus which occupies the last 7 cylinders of the recompiled area.

Appendix M. 3350 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3350 Directory Supplied with the Product Tape

Sample Directory for a 3350 Device

```
***************************************************************
*  3350    SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 3350 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS             *
***************************************************************
*
USER $ALLOC$ NOLOG
 MDISK A01 3350 000 001 VMSRES R
 MDISK C01 3350 000 001 VMPK01 R
 MDISK D01 3350 000 001 VMSTGE R
 MDISK E01 3350 000 001 PROFPK R
 MDISK F01 3350 000 001 SQLPK  R
*
USER $TEMP$   NOLOG
 MDISK A03 3350 234 043 VMSRES R
 MDISK B03 3350 278 040 VMSRES R
 MDISK C03 3350 227 100 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK A10 3350 318 100 VMSRES R
*
```

Figure 47 (Part 1 of 13). Sample Directory for a 3350 Device

```
USER $CPNUC$ NOLOG
 MDISK A09 3350 549 006 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A05 3350 101 002 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 3350 001 010 VMSRES R
 MDISK B02 3350 001 013 VMPK01 R
 MDISK A04 3350 001 002 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A06 3350 547 002 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A06 3350 277 001 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 3350 545 002 VMSRES R
*
**************************************************************
*             SYSTEM RELATED USERIDS
**************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 104 003 VMPK01 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 148 001 VMPK01 MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 3350 209 001 VMPK01 MR RBATCH WBATCH MBATCH
*
```

Figure 47 (Part 2 of 13). Sample Directory for a 3350 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 152 003 VMPK01 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 3350 107 001 VMPK01 MR RCPRM WCPRM
 MDISK 192 3350 108 007 VMPK01 MR ALL WCPRM
 MDISK 291 3350 115 001 VMPK01 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 395 005 VMPK01 MR RCSP WCSP MCSP
 MDISK 193 3350 340 006 VMPK01 MR RCSP WCSP MCSP
 MDISK 502 3350 406 020 VMPK01 MR RCSP WCSP MCSP
 MDISK 503 3350 426 020 VMPK01 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 3350 223 003 VMPK01 M RMOVR WMOVR MMOVR
*
```

Figure 47 (Part 3 of 13). Sample Directory for a 3350 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3350 135 003 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 193 3350 070 009 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 195 3350 138 009 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 123 3350 000 555 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 149 001 VMPK01 MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 3350 446 001 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 3350 226 001 VMPK01 MR RFSF1 WFSF1
*
```

Figure 47 (Part 4 of 13). Sample Directory for a 3350 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 3350 327 007 VMPK01 MR RFSF WFSF
 MDISK 192 3350 334 003 VMPK01 MR RFSF WFSF
 MDISK 193 3350 337 003 VMPK01 MR RFSF WFSF
 MDISK 194 3350 340 002 VMPK01 MR RFSF WFSF
 MDISK 195 3350 342 002 VMPK01 MR RFSF WFSF
 MDISK 197 3350 344 001 VMPK01 MR RFSF WFSF
 MDISK 200 3350 345 001 VMPK01 MR RFSF WFSF
 MDISK 400 3350 346 001 VMPK01 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 042 013 VMPK01 MR ALL WIIPS MIIPS
 MDISK 193 3350 055 015 VMPK01 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 3350 147 001 VMPK01 MR RPSR WPSR MPSR
*
```

Figure 47 (Part 5 of 13). Sample Directory for a 3350 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 3350 014 002 VMPK01 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 016 003 VMPK01 MR RISPF WISPF MISPF
 MDISK 192 3350 019 018 VMPK01 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M 16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3350  543 001 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M   4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3350 544 001 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 47 (Part 6 of 13). Sample Directory for a 3350 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 3350 000 555 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3350 000 555 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3350 000 555 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 128 3350 000 555 PROFPK MW RSYSRES WSYSRES MSYSRES
 MDISK 129 3350 000 555 SQLPK  MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3350 052 049 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 3350 011 007 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 3350 018 022 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 3350 493 014 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D 3350 418 028 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E 3350 194 028 VMSRES MW ALL     WMAINT MMAINT
 MDISK 201 3350 447 022 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 293 3350 469 012 VMSRES MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3350 481 012 VMSRES MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 3350 381 014 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 3350 129 005 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 310 3350 087 017 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 319 3350 149 045 VMSRES MW ALL     WMAINT MMAINT
 MDISK 31A 3350 126 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 324 3350 155 014 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 325 3350 197 009 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 390 3350 215 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 393 3350 001 072 VMSTGE WR RMAINT  WMAINT
 MDISK 394 3350 474 081 VMSTGE WR RMAINT  WMAINT
 MDISK 396 3350 073 033 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 3350 079 008 VMPK01 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF 3350 000 555 CEPACK MR READ WRITE
*
```

Figure 47 (Part 7 of 13). Sample Directory for a 3350 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 134 001 VMPK01 MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 117 001 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 3350 118 008 VMSRES MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 040 001 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT  250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3350 288 003 PROFPK MR RDBM WDBM MDBM
 MDISK 161 3350 291 010 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD 3350 325 012 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE 3350 313 012 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF 3350 301 012 PROFPK MR RDBM WDBM MDBM
*
```

Figure 47 (Part 8 of 13). Sample Directory for a 3350 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT  250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 3350 253 007 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 3350 260 003 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 116 004 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 3350 XXX 002 XXXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 191 3350 210 005 VMPK01 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3350 250 003 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 47 (Part 9 of 13). Sample Directory for a 3350 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3350 263 002 PROFPK MR RCAL WCAL MCAL
 MDISK 196 3350 265 009 PROFPK MR RCAL WCAL MCAL
*
***********************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
***********************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3350 347 007 VMPK01 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3350 001 010 SQLPK W
 MDISK 193 3350 011 023 SQLPK R  RSQL WSQL
 MDISK 195 3350 034 012 SQLPK RR RSQL WSQL MSQL
 MDISK 200 3350 046 029 SQLPK R  RSQL WSQL
 MDISK 201 3350 075 010 SQLPK R  RSQL WSQL
 MDISK 202 3350 085 096 SQLPK R  RSQL WSQL
*
USER SQLUSER SQLUSER 2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 3350 181 002 SQLPK W
*
```

Figure 47 (Part 10 of 13). Sample Directory for a 3350 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT 250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 3350 203 005 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 3350 208 010 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 3350 239 011 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 3350 274 014 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 206 003 VMPK01 MR RDUMP WDUMP MDUMP
 MDISK 123 3350 000 555 VMSRES RR
 MDISK 125 3350 000 555 VMPK01 RR
 MDISK 126 3350 000 555 VMSTGE RR
 MDISK 128 3350 000 555 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 037 005 VMPK01 MR RVMAP WVMAP MVMAP
*
```

Figure 47 (Part 11 of 13). Sample Directory for a 3350 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3350 356 007 VMPK01 W RBATCH WBATCH
 MDISK 192 3350 363 010 VMPK01 W RBATCH WBATCH
 MDISK 193 3350 373 007 VMPK01 W RBATCH WBATCH
 MDISK 194 3350 380 001 VMPK01 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 3350 354 002 VMPK01 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3350 150 002 VMPK01 MR RUTIL WUTIL MUTIL
*
```

Figure 47 (Part 12 of 13). Sample Directory for a 3350 Device

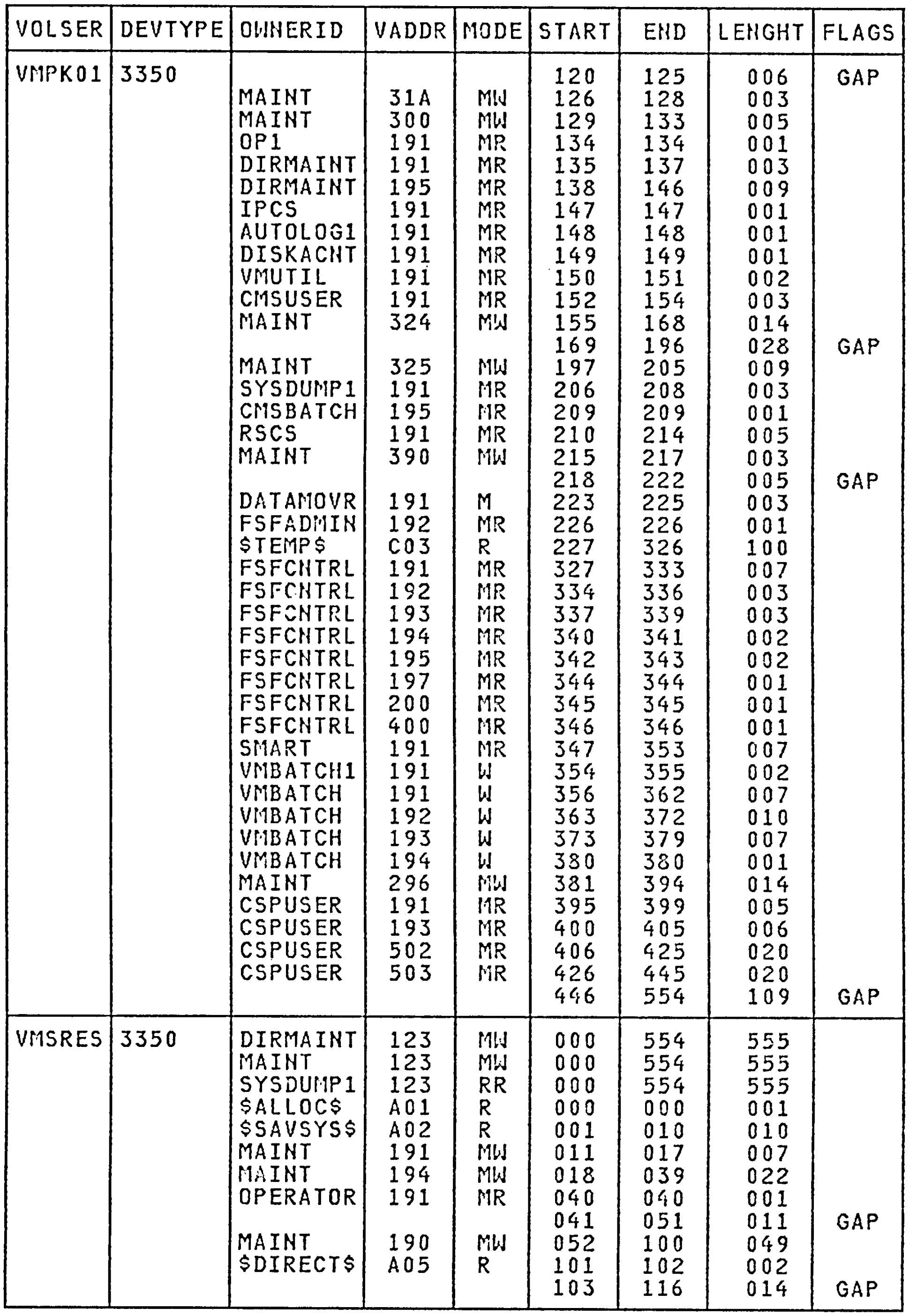
```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3350 SYSTEM
*MDISK 150 3350 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3350 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3350 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3350 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3350 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3350 SYSTEM
*MDISK 350 3350 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3350 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3350 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3350 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 00000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 00000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 00000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 47 (Part 13 of 13). Sample Directory for a 3350 Device

Map of a 3350 Directory Sample

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGHT	FLAGS
CEPACK	3350	OLTSEP	5FF	MR	000	554	555	
PROFPK	3350	MAINT	128	MW	000	554	555	
		SYSDUMP1	128	RR	000	554	555	
		$ALLOC$	E01	R	000	000	001	
		$SAVSYS$	A04	R	001	002	002	
					003	202	200	GAP
		SYSADMIN	191	MR	203	207	005	
		SYSADMIN	298	MR	208	217	010	
					218	238	021	GAP
		SYSADMIN	398	MR	239	249	011	
		SFBATCH	191	MR	250	252	003	
		PROMAIL	191	MR	253	259	007	
		PROMAIL	151	MR	260	262	003	
		SFCAL	191	MR	263	264	002	
		SFCAL	196	MR	265	273	009	
		SYSADMIN	399	MR	274	287	014	
		PRODBM	191	MR	288	290	003	
		PRODBM	161	MR	291	300	010	
		PRODBM	5FF	MR	301	312	012	
		PRODBM	5FE	MR	313	324	012	
		PRODBM	5FD	MR	325	336	012	
					337	554	218	GAP
SQLPK	3350	MAINT	129	MW	000	554	555	
		$ALLOC$	F01	R	000	000	001	
		SQLDBA	191	W	001	010	010	
		SQLDBA	193	R	011	033	023	
		SQLDBA	195	RR	034	045	012	
		SQLDBA	200	R	046	074	029	
		SQLDBA	201	R	075	084	010	
		SQLDBA	202	R	085	180	096	
		SQLUSER	191	W	181	182	002	
					183	554	372	GAP
VMPK01	3350	MAINT	125	MW	000	554	555	
		SYSDUMP1	125	RR	000	554	555	
		$ALLOC$	C01	R	000	000	001	
		$SAVSYS$	B02	R	001	013	013	
		ISMAINT	191	MR	014	015	002	
		ISPVM	191	MR	016	018	003	
		ISPVM	192	MR	019	036	018	
		VMAP	191	MR	037	041	005	
		IIPS	191	MR	042	054	013	
		IIPS	193	MR	055	069	015	
		DIRMAINT	193	MR	070	078	009	
		MAINT	3A0	MW	079	086	008	
		MAINT	310	MW	087	103	017	
		ADMIN	191	MR	104	106	003	
		CPRM	191	MR	107	107	001	
		CPRM	192	MR	108	114	007	
		CPRM	291	MR	115	115	001	
		PVM	191	MR	116	119	004	

Figure 48 (Part 1 of 3). Directory Map Sample for a 3350 Device



VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGHT	FLAGS
VMPK01	3350				120	125	006	GAP
		MAINT	31A	MW	126	128	003	
		MAINT	300	MW	129	133	005	
		OP1	191	MR	134	134	001	
		DIRMAINT	191	MR	135	137	003	
		DIRMAINT	195	MR	138	146	009	
		IPCS	191	MR	147	147	001	
		AUTOLOG1	191	MR	148	148	001	
		DISKACNT	191	MR	149	149	001	
		VMUTIL	191	MR	150	151	002	
		CMSUSER	191	MR	152	154	003	
		MAINT	324	MW	155	168	014	
					169	196	028	GAP
		MAINT	325	MW	197	205	009	
		SYSDUMP1	191	MR	206	208	003	
		CMSBATCH	195	MR	209	209	001	
		RSCS	191	MR	210	214	005	
		MAINT	390	MW	215	217	003	
					218	222	005	GAP
		DATAMOVR	191	M	223	225	003	
		FSFADMIN	192	MR	226	226	001	
		$TEMP$	C03	R	227	326	100	
		FSFCNTRL	191	MR	327	333	007	
		FSFCNTRL	192	MR	334	336	003	
		FSFCNTRL	193	MR	337	339	003	
		FSFCNTRL	194	MR	340	341	002	
		FSFCNTRL	195	MR	342	343	002	
		FSFCNTRL	197	MR	344	344	001	
		FSFCNTRL	200	MR	345	345	001	
		FSFCNTRL	400	MR	346	346	001	
		SMART	191	MR	347	353	007	
		VMBATCH1	191	W	354	355	002	
		VMBATCH	191	W	356	362	007	
		VMBATCH	192	W	363	372	010	
		VMBATCH	193	W	373	379	007	
		VMBATCH	194	W	380	380	001	
		MAINT	296	MW	381	394	014	
		CSPUSER	191	MR	395	399	005	
		CSPUSER	193	MR	400	405	006	
		CSPUSER	502	MR	406	425	020	
		CSPUSER	503	MR	426	445	020	
					446	554	109	GAP
VMSRES	3350	DIRMAINT	123	MW	000	554	555	
		MAINT	123	MW	000	554	555	
		SYSDUMP1	123	RR	000	554	555	
		$ALLOC$	A01	R	000	000	001	
		$SAVSYS$	A02	R	001	010	010	
		MAINT	191	MW	011	017	007	
		MAINT	194	MW	018	039	022	
		OPERATOR	191	MR	040	040	001	
					041	051	011	GAP
		MAINT	190	MW	052	100	049	
		$DIRECT$	A05	R	101	102	002	
					103	116	014	GAP

Figure 48 (Part 2 of 3). Directory Map Sample for a 3350 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGHT	FLAGS
VMSRES	3350	OPERATNS	191	MR	117	117	001	
		OPERATNS	193	MR	118	125	008	
					126	148	023	GAP
		MAINT	319	MW	149	193	045	
		MAINT	19E	MW	194	221	028	
					222	233	012	GAP
		$TEMP$	A03	R	234	276	043	
		$SYSCKP$	A06	R	277	277	001	
		$TEMP$	B03	R	278	317	040	
		$TDISK$	A10	R	318	417	100	
		MAINT	19D	MW	418	445	028	
		EREP	191	WR	446	446	001	
		MAINT	201	MW	447	468	022	
		MAINT	293	MW	469	480	012	
		MAINT	294	MW	481	492	012	
		MAINT	196	MW	493	506	014	
					507	542	036	GAP
		IVPM1	191	WR	543	543	001	
		IVPM2	191	WR	544	544	001	
		$SYSWRM$	A07	R	545	546	002	
		$SYSERR$	A06	R	547	548	002	
		$CPNUC$	A09	R	549	554	006	
VMSTGE	3350	MAINT	126	MW	000	554	555	
		SYSDUMP1	126	RR	000	554	555	
		$ALLOC$	D01	R	000	000	001	
		MAINT	393	WR	001	072	072	
		MAINT	396	WR	073	105	033	
					106	473	368	GAP
		MAINT	394	WR	474	554	081	

Figure 48 (Part 3 of 3). Directory Map Sample for a 3350 Device

3350 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3350    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
         COPY  OPTIONS
***********************************************************************
*
*       C L U S T E R    A N D    L I N E    C O N T R O L L E R S
*
***********************************************************************
         SPACE
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
       SPACE
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 49 (Part 1 of 4). Sample DMKRIO File for a 3350 Device

```
**********************************************************************
*
*                     C H A N N E L    Z E R O
*
* NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4331
*           OR 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*           016 MAY BE USED AS DISPLAY CONSOLE ON 168 AND 165-II.
*
**********************************************************************
         SPACE
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=A
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
GRAF     RDEVICE ADDRESS=(018,2),DEVTYPE=3278,MODEL=2
CONS     RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(030,16),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
**********************************************************************
*
*                     C H A N N E L    O N E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,32),DEVTYPE=3277
         RDEVICE ADDRESS=(140,16),DEVTYPE=3350
         RDEVICE ADDRESS=(150,16),DEVTYPE=3350
         RDEVICE ADDRESS=(160,16),DEVTYPE=3350
         RDEVICE ADDRESS=(170,16),DEVTYPE=3350
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1C0,16),DEVTYPE=3350
         EJECT
```

Figure 49 (Part 2 of 4). Sample DMKRIO File for a 3350 Device

```
**********************************************************************
*
*                     C H A N N E L    T W O
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(220,32),DEVTYPE=3277
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3350
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         EJECT
**********************************************************************
*
*                     C H A N N E L    T H R E E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3350
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         EJECT
**********************************************************************
*
*                     C H A N N E L    F O U R
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3350
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
**********************************************************************
*
*                     C H A N N E L    F I V E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
         EJECT
```

Figure 49 (Part 3 of 4). Sample DMKRIO File for a 3350 Device

```
***********************************************************************
*
*                     C O N T R O L    U N I T S
*
***********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=32-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=160,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU2   RCTLUNIT ADDRESS=220,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803
         RCTLUNIT ADDRESS=580,CUTYPE=3803
         EJECT
***********************************************************************
*
*                         C H A N N E L S
*
***********************************************************************
         SPACE
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
***********************************************************************
*
*                C O N S O L E
*
***********************************************************************
         SPACE
         RIOGEN  CONS=014,ALTCONS=(01F,009,01A,016,010)
         END
```

Figure 49 (Part 4 of 4). Sample DMKRIO File for a 3350 Device

3350 DMKSYS File Supplied With The Product Tape

The 3350 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=3350,                                           X
               SYSCLR=YES,                                             X
               SYSNUC=549,                                             X
               SYSWRM=(545,2),                                         X
               SYSERR=(547,2),                                         X
               SYSCKP=(277,1)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 50. Sample DMKSYS File for a 3350 Device

3350 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT       VM/SP REL 3       3350 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSCYL=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSCYL  - THE CYLINDER ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 51 (Part 1 of 8). Sample DMKSNT File for a 3350 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSCYL=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
*******************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
*******************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*******************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*     ( THE ALLOCATIONS ARE BASED ON 120 PAGES/3350 CYLINDER )
*          CYL 1, PAGE 1 TO CYL 2, PAGE 18 (138 PAGES)
*         137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*******************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(001,1),                                        X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=52,                                              X
               VSYSRES=VMSRES
         EJECT
```

Figure 51 (Part 2 of 8). Sample DMKSNT File for a 3350 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 2, PAGE 19 TO CYL 3, PAGE 36 (138 PAGES)
********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(002,19),                                       X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=52,                                              X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 37 TO CYL 3, PAGE 53 (17 PAGES)
********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(003,37),                                       X
               SYSPGNM=(784-799),                                      X
               SYSPGCT=16,                                             X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 54 TO CYL 3, PAGE 102 (49 PAGES)
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(003,54),                                       X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 51 (Part 3 of 8). Sample DMKSNT File for a 3350 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 103 TO CYL 4, PAGE 95 ( 113 PAGES )
********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(03,103),                                       X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 4, PAGE 96 TO CYL 5, PAGE 120 (145 PAGES)
********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(004,96),                                       X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 6, PAGE 001 TO CYL 6, PAGE 17 (17 PAGES)
********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(06,001),                                       X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 51 (Part 4 of 8). Sample DMKSNT File for a 3350 Device

```
********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 6, PAGE 18 TO CYL 6, PAGE 34 (17 PAGES)
********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(006,018),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
********************************************************************
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 6, PAGE 35 TO CYL 6, PAGE 51 (17 PAGES)
********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(006,035),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 06, PAGE 52 TO CYL 07, PAGE 28 (97 PAGES)
********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                      X
               SYSSIZE=384K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(006,052),                                      X
               SYSPGCT=96,                                             X
               SYSPGNM=(1072-1167),                                    X
               SYSHRSG=(67,68,69,70,71,72),                            X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 51 (Part 5 of 8). Sample DMKSNT File for a 3350 Device

```
***********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 7, PAGE  29 TO CYL 7, PAGE 45  (17 PAGES)
***********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(007,029),                                      X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
***********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 7, PAGE 46 TO CYL 8, PAGE  22 (97 PAGES)
***********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(007,46),                                       X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
***********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 8, PAGE  23 TO CYL 8, PAGE 119 (97 PAGES)
***********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(08,023),                                       X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 51 (Part 6 of 8). Sample DMKSNT File for a 3350 Device

```
********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 8, PAGE 120 TO CYL 9, PAGE 64  (65 PAGES)
********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(008,120),                                      X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
* >        CYL 1, PAGE 1 TO CYL 1, PAGE 97 (97 PAGES)
********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(001,01),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 4, PAGE 29 TO CYL 5, PAGE  5 (97 PAGES)
********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(004,29),SYSPGCT=96,              X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 5, PAGE 6 TO CYL 5, PAGE 54 (49 PAGES)
********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(005,006),                        X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 51 (Part 7 of 8). Sample DMKSNT File for a 3350 Device

```
********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 5, PAGE 55 TO CYL 7, PAGE 55 (241 PAGES)
********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                    X
               SYSVOL=VMPK01,                                         X
               SYSSIZE=960K,                                          X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                         X
               SYSPGNM=(1808-2047),                                   X
               SYSPGCT=240,                                           X
               SYSSTRT=(05,55),                                       X
               VSYSRES=,                                              X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 7, PAGE 56 TO CYL 11, PAGE 24 (449 PAGES)
********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                    X
               SYSVOL=VMPK01,                                         X
               SYSSIZE=1024K,                                         X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                             X
               SYSPGNM=(2336-2783),                                   X
               SYSPGCT=448,                                           X
               SYSSTRT=(007,56),                                      X
               VSYSRES=,                                              X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE 73  (193 PAGES)
********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                      X
               SYSSIZE=384K,                                          X
               VSYSRES=,                                              X
               VSYSADR=IGNORE,                                        X
               SYSCYL=,                                               X
               SYSVOL=PROFPK,                                         X
               SYSSTRT=(01,01),                                       X
               SYSPGCT=192,                                           X
               SYSPGNM=(1328-1519),                                   X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 51 (Part 8 of 8). Sample DMKSNT File for a 3350 Device

VMSRES Pack Layout for 3350 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-17	7	191 minidisk for MAINT
18-39	22	194 minidisk for MAINT (CP)
40	1	191 minidisk for OPERATOR
41-51	11	xxx minidisk for user option
52-100	49	190 minidisk for MAINT (CMS system disk)*
101-102	2	VM/SP directory
103-116	14	xxx minidisk for user option
117	1	191 minidisk for OPERATNS
118-125	8	193 minidisk for OPERATNS
126-148	23	xxx minidisk for user option
149-193	45	319 minidisk for MAINT
194-221	28	19E minidisk for MAINT
222-233	12	xxx minidisk for user option
234-276	43	Spool space
277	1	Spool file checkpoint
278-317	40	Spool space
318-417	100	T-DISK space
418-445	28	19D minidisk for MAINT (HELP)
446	1	191 minidisk for EREP
447-468	22	201 minidisk for MAINT
469-480	12	293 minidisk for MAINT (CMS service)
481-492	12	294 minidisk for MAINT (CP service)
493-542	50	xxx minidisk for user option
543	1	191 minidisk for IVPM1
544	1	191 minidisk for IVPM2
545-546	2	Warm start data
547-548	2	I/O Error Recording area
549-554	6	CP nucleus

Figure 52. VMSRES Layout for a 3350 Starter System

* Specify cylinder 45 to load the CMS nucleus which occupies the first 2 cylinders of the recompiled area.
Specify cylinder 47 to load the CMSL nucleus which occupies the last 2 cylinders of the recompiled area.

Appendix N. 3370 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3370 Directory Supplied with the Product Tape

Sample Directory for a FB-512 3370 Device

```
***************************************************************
*  3370      SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 FB-512 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$ NOLOG
 MDISK A01 FB-512 000000 000016 VMSRES R
 MDISK C01 FB-512 000000 000016 VMPK01 R
 MDISK D01 FB-512 000000 000016 VMSTGE R
 MDISK E01 FB-512 000000 000016 PROFPK R
 MDISK F01 FB-512 000000 000016 SQLPK  R
*
USER $TEMP$ NOLOG
 MDISK A06 FB-512 094816 027384 VMSRES R
 MDISK A07 FB-512 126016 150000 VMSRES R
 MDISK C01 FB-512 215672 080000 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK A03 FB-512 115672 100000 VMPK01 R
 MDISK B03 FB-512 380328 035000 VMSRES R
*
```

Figure 53 (Part 1 of 13). Sample Directory for a 3370 Device

```
USER $CPNUC$ NOLOG
 MDISK A10 FB-512 123736 002280 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A05 FB-512 092480 001824 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 FB-512 000016 009360 VMSRES R
 MDISK C02 FB-512 000016 010450 VMPK01 R
 MDISK A04 FB-512 000016 001544 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A09 FB-512 122712 000512 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A07 FB-512 123224 000512 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 FB-512 122200 000512 VMSRES R
*
****************************************************************
*             SYSTEM RELATED USERIDS
****************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 347738 002280 VMSRES MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 350018 000200 VMSRES MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 FB-512 304288 000960 VMSRES MR RBATCH WBATCH MBATCH
*
```

Figure 53 (Part 2 of 13). Sample Directory for a 3370 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 350218 002280 VMSRES MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 FB-512 340858 000200 VMSRES MR RCPRM WCPRM
 MDISK 192 FB-512 341058 006000 VMSRES MR ALL WCPRM
 MDISK 291 FB-512 347058 000680 VMSRES MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 474312 004000 VMPK01 MR RCSP WCSP MCSP
 MDISK 193 FB-512 478312 005852 VMPK01 MR RCSP WCSP MCSP
 MDISK 502 FB-512 484164 018810 VMPK01 MR RCSP WCSP MCSP
 MDISK 503 FB-512 502974 018810 VMPK01 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 FB-512 320368 002090 VMSRES M RMOVR WMOVR MMOVR
*
```

Figure 53 (Part 3 of 13). Sample Directory for a 3370 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 FB-512 315808 002280 VMSRES MR RDIRM WDIRM MDIRM
 MDISK 193 FB-512 355828 007752 VMSRES MR RDIRM WDIRM MDIRM
 MDISK 195 FB-512 305248 007800 VMSRES MR RDIRM WDIRM MDIRM
 MDISK 123 FB-512 000000 558000 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 314872 000480 VMSRES MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 FB-512 276016 000912 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 FB-512 427306 000836 VMPK01 MR RFSF1 WFSF1
*
```

Figure 53 (Part 4 of 13). Sample Directory for a 3370 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 FB-512 428142 006270 VMPK01 MR RFSF WFSF
 MDISK 192 FB-512 434412 002090 VMPK01 MR RFSF WFSF
 MDISK 193 FB-512 436502 002090 VMPK01 MR RFSF WFSF
 MDISK 194 FB-512 438592 001254 VMPK01 MR RFSF WFSF
 MDISK 195 FB-512 439846 001254 VMPK01 MR RFSF WFSF
 MDISK 197 FB-512 441100 000418 VMPK01 MR RFSF WFSF
 MDISK 200 FB-512 441518 000418 VMPK01 MR RFSF WFSF
 MDISK 400 FB-512 441936 000418 VMPK01 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 380072 010776 VMPK01 MR ALL WIIPS MIIPS
 MDISK 193 FB-512 084350 016416 VMPK01 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 FB-512 315352 000456 VMSRES MR RPSR WPSR MPSR
*
```

Figure 53 (Part 5 of 13). Sample Directory for a 3370 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 FB-512 352498 002000 VMSRES MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 354498 001330 VMSRES MR RISPF WISPF MISPF
 MDISK 192 FB-512 100766 014630 VMPK01 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M 16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512  094304 000256 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M 4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 094560 000256 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 53 (Part 6 of 13). Sample Directory for a 3370 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 FB-512 000000 558000 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 125 FB-512 000000 558000 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 FB-512 000000 558000 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 190 FB-512 046912 045568 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 FB-512 009376 005282 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 FB-512 014658 020520 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 FB-512 276928 012540 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D FB-512 417608 020900 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E FB-512 338272 025080 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 201 FB-512 047376 020520 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 293 FB-512 010466 010032 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 FB-512 074736 009614 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 FB-512 289468 012540 VMSRES MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 FB-512 322458 003800 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 310 FB-512 326258 014600 VMSRES MW ALL     WMAINT MMAINT
 MDISK 319 FB-512 295672 042600 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 31A FB-512 302008 002280 VMSRES MW ALL     WMAINT MMAINT
 MDISK 324 FB-512 027914 012540 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 325 FB-512 036202 009120 VMSRES MW ALL     WMAINT MMAINT
 MDISK 390 FB-512 045322 001152 VMSRES MW ALL     WMAINT MMAINT
 MDISK 393 FB-512 000032 068166 VMSTGE WR RMAINT  WMAINT
 MDISK 394 FB-512 068198 071064 VMSTGE WR RMAINT  WMAINT
 MDISK 396 FB-512 139262 029260 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 FB-512 373672 006400 VMPK01 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF FB-512 000000 558000 CEPACK MR READ WRITE
*
```

Figure 53 (Part 7 of 13). Sample Directory for a 3370 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 046474 000400 VMSRES MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 366856 000456 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 FB-512 367312 007296 VMSRES MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 035178 001024 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT 250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 075042 002508 PROFPK MR RDBM WDBM MDBM
 MDISK 161 FB-512 077550 008360 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD FB-512 105974 010032 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE FB-512 095942 010032 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF FB-512 085910 010032 PROFPK MR RDBM WDBM MDBM
*
```

Figure 53 (Part 8 of 13). Sample Directory for a 3370 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT 250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 FB-512 045486 006270 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 FB-512 051756 002508 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 025058 002400 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 FB-512 XXXXXX XXXXXX VMXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 318088 002280 VMSRES MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 042978 002508 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 53 (Part 9 of 13). Sample Directory for a 3370 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 FB-512 054264 001254 PROFPK MR RCAL WCAL MCAL
 MDISK 196 FB-512 055518 007524 PROFPK MR RCAL WCAL MCAL
*
***********************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
***********************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 FB-512 442354 006688 VMPK01 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 FB-512 000016 008100 SQLPK W
 MDISK 193 FB-512 008116 018630 SQLPK R  RSQL WSQL
 MDISK 195 FB-512 026746 009720 SQLPK RR RSQL WSQL MSQL
 MDISK 200 FB-512 036466 023490 SQLPK R  RSQL WSQL
 MDISK 201 FB-512 059956 009600 SQLPK R  RSQL WSQL
 MDISK 202 FB-512 069556 092160 SQLPK R  RSQL WSQL
*
USER SQLUSER SQLUSER 2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 FB-512 161716 001620 SQLPK W
*
```

Figure 53 (Part 10 of 13). Sample Directory for a 3370 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT 250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 FB-512 001560 004000 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 FB-512 005560 008980 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 FB-512 033478 009500 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 FB-512 063042 012000 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 449042 002280 VMPK01 MR RDUMP WDUMP MDUMP
 MDISK 123 FB-512 000000 558000 VMSRES RR
 MDISK 125 FB-512 000000 558000 VMPK01 RR
 MDISK 126 FB-512 000000 558000 VMSTGE RR
 MDISK 128 FB-512 000000 558000 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 020498 004560 VMPK01 MR RVMAP WVMAP MVMAP
*
```

Figure 53 (Part 11 of 13). Sample Directory for a 3370 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 FB-512 451322 006270 VMPK01 W RBATCH WBATCH
 MDISK 192 FB-512 457592 008360 VMPK01 W RBATCH WBATCH
 MDISK 193 FB-512 465952 006270 VMPK01 W RBATCH WBATCH
 MDISK 194 FB-512 472222 000418 VMPK01 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 FB-512 472640 001672 VMPK01 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 FB-512 313048 001824 VMSRES MR RUTIL WUTIL MUTIL
*
```

Figure 53 (Part 12 of 13). Sample Directory for a 3370 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* FB-512 SYSTEM
*MDISK 150 3330 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3330 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3330 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3330 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3330 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3350 SYSTEM
*MDISK 350 3350 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3350 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3350 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3350 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 000000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 000000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 000000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 53 (Part 13 of 13). Sample Directory for a 3370 Device

Map of a 3370 Directory Sample

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	FB-512	OLTSEP	5FF	MR	000	557999	558000	
PROFPK	FB-512	SYSDUMP1	128	RR	000	557999	558000	
		$ALLOC$	E01	R	000	015	016	
		$SAVSYS$	A04	R	016	1559	1544	
		SYSADMIN	191	MR	1560	5559	4000	
		SYSADMIN	298	MR	5560	14539	8980	
					14540	33477	18938	GAP
		SYSADMIN	398	MR	33478	42977	9500	
		SFBATCH	191	MR	42978	45485	2508	
		PROMAIL	191	MR	45486	51755	6270	
		PROMAIL	151	MR	51756	54263	2508	
		SFCAL	191	MR	54264	55517	1254	
		SFCAL	196	MR	55518	63041	7524	
		SYSADMIN	399	MR	63042	75041	12000	
		PRODBM	191	MR	75042	77549	2508	
		PRODBM	161	MR	77550	85909	8360	
		PRODBM	5FF	MR	85910	95941	10032	
		PRODBM	5FE	MR	95942	105973	10032	
		PRODBM	5FD	MR	105974	116005	10032	
					116006	557999	441994	GAP
SQLPK	FB-512	$ALLOC$	F01	R	000	015	016	
		SQLDBA	191	W	016	8115	8100	
		SQLDBA	193	R	8116	26745	18630	
		SQLDBA	195	RR	26746	36465	9720	
		SQLDBA	200	R	36466	59955	23490	
		SQLDBA	201	R	59956	69555	9600	
		SQLDBA	202	R	69556	161715	92160	
		SQLUSER	191	W	161716	163335	1620	
					163336	557999	394664	GAP
VMPK01	FB-512	MAINT	125	MW	000	557999	558000	
		SYSDUMP1	125	RR	000	557999	558000	
		$ALLOC$	C01	R	000	015	016	
		$SAVSYS$	C02	R	016	10465	10450	
		MAINT	293	MW	10466	20497	10032	
		VMAP	191	MR	20498	25057	4560	
		PVM	191	MR	25058	27457	2400	
					27458	27913	456	GAP
		MAINT	324	MW	27914	40453	12540	
					40454	47375	6922	GAP
		MAINT	201	MW	47376	67895	20520	
					67896	74735	6840	GAP
		MAINT	294	MW	74736	84349	9614	
					84350	100765	16416	GAP
		ISPVM	192	MR	100766	115395	14630	
					115396	115671	276	GAP
		$TDISK$	A03	R	115672	215671	100000	
		$TEMP$	C01	R	215672	295671	80000	
		MAINT	319	MW	295672	338271	42600	
		MAINT	19E	MW	338272	363351	25080	
		IIPS	192	MR	363352	373671	10320	
		MAINT	3A0	MW	373672	380071	6400	

Figure 54 (Part 1 of 3). Directory Map Sample for a 3370 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK01	FB-512	IIPS	191	MR	380072	390847	10776	
					390848	427305	36458	GAP
		FSFADMIN	192	MR	427306	428141	836	
		FSFCNTRL	191	MR	428142	434411	6270	
		FSFCNTRL	192	MR	434412	436501	2090	
		FSFCNTRL	193	MR	436502	438591	2090	
		FSFCNTRL	194	MR	438592	439845	1254	
		FSFCNTRL	195	MR	439846	441099	1254	
		FSFCNTRL	197	MR	441100	441517	418	
		FSFCNTRL	200	MR	441518	441935	418	
		FSFCNTRL	400	MR	441936	442353	418	
		SMART	191	MR	442354	449041	6688	
		SYSDUMP1	191	MR	449042	451321	2280	
		VMBATCH	191	W	451322	457591	6270	
		VMBATCH	192	W	457592	465951	8360	
		VMBATCH	193	W	465952	472221	6270	
		VMBATCH	194	W	472222	472639	418	
		VMBATCH1	191	W	472640	474311	1672	
		CSPUSER	191	MR	474312	478311	4000	
		CSPUSER	193	MR	478312	484163	5852	
		CSPUSER	502	MR	484164	502973	18810	
		CSPUSER	503	MR	502974	521783	18810	
					521784	557999	36216	GAP
VMSRES	FB-512	DIRMAINT	123	MW	000	557999	558000	
		MAINT	123	MW	000	557999	558000	
		SYSDUMP1	123	RR	000	557999	558000	
		$ALLOC$	A01	R	000	015	016	
		$SAVSYS$	A02	R	016	9375	9360	
		MAINT	191	MW	9376	14657	5282	
		MAINT	194	MW	14658	35177	20520	
		OPERATOR	191	MR	35178	36201	1024	
		MAINT	325	MW	36202	45321	9120	
		MAINT	390	MW	45322	46473	1152	
		OP1	191	MR	46474	46873	400	
					46874	46911	038	GAP
		MAINT	190	MW	46912	92479	45568	
		$DIRECT$	A05	R	92480	94303	1824	
		IVPM1	191	WR	94304	94559	256	
		IVPM2	191	WR	94560	94815	256	
		$TEMP$	A06	R	94816	122199	27384	
		$SYSWRM$	A07	R	122200	122711	512	
		$SYSERR$	A09	R	122712	123223	512	
		$SYSCKP$	A07	R	123224	123735	512	
		$CPNUC$	A10	R	123736	126015	2280	
		$TEMP$	A07	R	126016	276015	150000	
		EREP	191	WR	276016	276927	912	
		MAINT	196	MW	276928	289467	12540	
		MAINT	296	MW	289468	302007	12540	
		MAINT	31A	MW	302008	304287	2280	
		CMSBATCH	195	MR	304288	305247	960	
		DIRMAINT	195	MR	305248	313047	7800	
		VMUTIL	191	MR	313048	314871	1824	
		DISKACNT	191	MR	314872	315351	480	
		IPCS	191	MR	315352	315807	456	
		DIRMAINT	191	MR	315808	318087	2280	
		RSCS	191	MR	318088	320367	2280	

Figure 54 (Part 2 of 3). Directory Map Sample for a 3370 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSRES	FB-512	DATAMOVR	191	M	320368	322457	2090	
		MAINT	300	MW	322458	326257	3800	
		MAINT	310	MW	326258	340857	14600	
		CPRM	191	MR	340858	341057	200	
		CPRM	192	MR	341058	347057	6000	
		CPRM	291	MR	347058	347737	680	
		ADMIN	191	MR	347738	350017	2280	
		AUTOLOG1	191	MR	350018	350217	200	
		CMSUSER	191	MR	350218	352497	2280	
		ISMAINT	191	MR	352498	354497	2000	
		ISPVM	191	MR	354498	355827	1330	
		DIRMAINT	193	MR	355828	363579	7752	
					363580	366855	3276	GAP
		OPERATNS	191	MR	366856	367311	456	
		OPERATNS	193	MR	367312	374607	7296	
					374608	380327	5720	GAP
		$TDISK$	B03	R	380328	415327	35000	
					415328	417607	2280	GAP
		MAINT	19D	MW	417608	438507	20900	
					438508	557999	119492	GAP
VMSTGE	FB-512	MAINT	126	MW	000	557999	558000	
		SYSDUMP1	126	RR	000	557999	558000	
		$ALLOC$	D01	R	000	015	016	
					016	031	016	GAP
		MAINT	393	WR	032	68197	68166	
		MAINT	394	WR	68198	139261	71064	
		MAINT	396	WR	139262	168521	29260	
					168522	557999	389478	GAP

Figure 54 (Part 3 of 3). Directory Map Sample for a 3370 Device

3370 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3370    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
         COPY  OPTIONS
***********************************************************************
*
*       C L U S T E R    A N D    L I N E    C O N T R O L L E R S
*
***********************************************************************
         SPACE
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
       SPACE
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 55 (Part 1 of 4). Sample DMKRIO File for a 3370 Device

```
**********************************************************************
*
*                     C H A N N E L    Z E R O
*
*    NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4331 OR
*          THE 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*           016 MAY BE USED AS DISPLAY CONSOLE ON 168 AND 165-II.
*
**********************************************************************
         SPACE
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=A
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
GRAF     RDEVICE ADDRESS=(018,2),DEVTYPE=3278,MODEL=2
CONS     RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(030,16),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
**********************************************************************
*
*                     C H A N N E L    O N E
*
**********************************************************************
         SPACE
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(128,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(130,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(138,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(160,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(168,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(170,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(178,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1A0,16),DEVTYPE=3277
         EJECT
```

Figure 55 (Part 2 of 4). Sample DMKRIO File for a 3370 Device

```
***********************************************************************
*
*                     C H A N N E L    T W O
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(220,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(230,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3350
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2A0,16),DEVTYPE=3277
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         EJECT
***********************************************************************
*
*                     C H A N N E L    T H R E E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(300,2),DEVTYPE=8809
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3350
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         EJECT
***********************************************************************
*
*                     C H A N N E L    F O U R
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3350
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
***********************************************************************
*
*                     C H A N N E L    F I V E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
         EJECT
```

Figure 55 (Part 3 of 4). Sample DMKRIO File for a 3370 Device

```
***********************************************************************
*
*                     C O N T R O L    U N I T S
*
***********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0A0,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=16-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=160,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1A0,CUTYPE=3272,FEATURE=16-DEVICE
RCTLU2   RCTLUNIT ADDRESS=220,CUTYPE=FTA
         RCTLUNIT ADDRESS=230,CUTYPE=FTA
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2A0,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=300,CUTYPE=FTA
         RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803
         RCTLUNIT ADDRESS=580,CUTYPE=3803
         EJECT
***********************************************************************
*
*                       C H A N N E L S
*
***********************************************************************
         SPACE
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
***********************************************************************
*
*                 C O N S O L E    D E F I N I T I O N
*
***********************************************************************
         SPACE
         RIOGEN   CONS=010,ALTCONS=(01F,009,01A,016,014)
         END
```

Figure 55 (Part 4 of 4). Sample DMKRIO File for a 3370 Device

3370 DMKSYS File Supplied With The Product Tape

The 3370 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=FB-512,                                         X
               SYSCLR=YES,                                             X
               SYSNUC=15467,                                           X
               SYSWRM=(15275,64),                                      X
               SYSERR=(15339,64),                                      X
               SYSCKP=(15403,64)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 56. Sample DMKSYS File for a 3370 Device

3370 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT       VM/SP REL 3     FB-512 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSBLOK=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSBLOK - THE BLOCK    ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
*
```

Figure 57 (Part 1 of 8). Sample DMKSNT File for a 3370 Device

```
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSBLOK=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
**********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
**********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 1119 (PAGE 139)
*      137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*     TOTAL = 138 PAGES
*
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(2),                                            X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSBLOK=46912,                                          X
               VSYSRES=VMSRES
         EJECT
```

Figure 57 (Part 2 of 8). Sample DMKSNT File for a 3370 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 1120 (PAGE 140) TO BLK 2223 (PAGE 277)
*     TOTAL = 138 PAGES
********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(140),                                          X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSBLOK=46912,                                          X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2224 (PAGE 278) TO BLK 2359 (PAGE 294)
*     TOTAL = 17 PAGES
********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(278),                                          X
               SYSPGCT=16,                                             X
               SYSPGNM=(784-799),                                      X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               VSYSRES=,                                               X
               SYSBLOK=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2360 (PAGE 295) TO BLK 2751 (PAGE 343)
*     TOTAL = 49 PAGES
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(295),                                          X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 57 (Part 3 of 8). Sample DMKSNT File for a 3370 Device

```
***********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 2752 (PAGE 344) TO BLK 3655 (PAGE 456)
*     TOTAL = 113 PAGES
***********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(344),                                          X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
***********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 3656 (PAGE 457) TO BLK 4815 (PAGE 601)
*     TOTAL = 145 PAGES
***********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(457),                                          X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
***********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 4816 (PAGE 602) TO BLK 4951 (PAGE 618)
*     TOTAL = 17 PAGES
***********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(602),                                          X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSBLOK=,                                               X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 57 (Part 4 of 8). Sample DMKSNT File for a 3370 Device

```
*********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 4952 (PAGE 619) TO BLK 5087 (PAGE 635)
*     TOTAL = 17 PAGES
**********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                            X
               CPSIZE=48K,                                               X
               CPTYPE=EP,                                                X
               SYSSTRT=(619),                                            X
               SYSPGCT=16,                                               X
               SYSVOL=VMSRES
         EJECT
**********************************************************************
*
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 5088 (PAGE 636) TO BLK 5223 (PAGE 652)
*     TOTAL = 17 PAGES
*
**********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                            X
               CPSIZE=48K,                                               X
               CPTYPE=EP,                                                X
               SYSSTRT=(636),                                            X
               SYSPGCT=16,                                               X
               SYSVOL=VMSRES
         EJECT
***********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 5224 (PAGE 653) TO BLK 5999 (PAGE 749)
*     TOTAL = 97 PAGES
**********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                        X
               SYSSIZE=384K,                                             X
               SYSVOL=VMSRES,                                            X
               SYSCYL=,                                                  X
               SYSSTRT=(653),                                            X
               SYSPGCT=96,                                               X
               SYSPGNM=(1072-1167),                                      X
               SYSHRSG=(67,68,69,70,71,72),                              X
               VSYSRES=,                                                 X
               VSYSADR=IGNORE
         EJECT
```

Figure 57 (Part 5 of 8). Sample DMKSNT File for a 3370 Device

```
*********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6000 (PAGE 750) TO BLK 6135 (PAGE 766)
*     TOTAL = 17 PAGES
*********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(750),                                          X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6136 (PAGE 767) TO BLK 6911 (PAGE 863)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(767),                                          X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
*********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*       FB-512 BLK 6912 (PAGE 864) TO BLK 7687 (PAGE 960)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(864),                                          X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 57 (Part 6 of 8). Sample DMKSNT File for a 3370 Device

```
*********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*       FB-512 BLK 7688 (PAGE 961) TO BLK 8207 (PAGE 1025)
*     TOTAL = 65 PAGES
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(961),                                          X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 791 (PAGE 98)
*     TOTAL = 97 PAGES
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(002),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 3120 (PAGE 390) TO BLK 3895 (PAGE 486)
*     TOTAL = 97 PAGES
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(390),SYSPGCT=96,                 X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 3896 (PAGE 487) TO BLK 4287 (PAGE 535)
*     TOTAL = 49 PAGES
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(487),                            X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 57 (Part 7 of 8). Sample DMKSNT File for a 3370 Device

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 4288 (PAGE 536) TO BLK 6215 (PAGE 776)
*     TOTAL = 241 PAGES
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(536),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 6216 (PAGE 777) TO BLK 9807 (PAGE 1225)
*     TOTAL = 449 PAGES
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(777),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 1559 (PAGE 194)
*     TOTAL = 193 PAGES
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(2),                                            X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 57 (Part 8 of 8). Sample DMKSNT File for a 3370 Device

VMSRES Pack Layout for 3370 Starter System

Block No. (Directory use)	No. of FB-512 Blocks	Page Number (CP Format use)	Number of Pages (BLK/8)	Contents
0-15	16	0-1	2	Allocation
16-9375	9360	2-1171	1170	Saved segments
9376-14657	5282	1172-	**	191 minidisk for MAINT
14658-35177	20520		**	194 minidisk for MAINT (CP)
35178-36201	1024		**	191 minidisk for OPERATOR
36202-46911	10710	-5863	**	xxx minidisk space for user option
46912-92479	45568	5864-11559	5696	190 MAINT (CMS system minidisk) *
92480-94303	1824	11560-11787	228	VM/SP directory
94304-94559	256	11788-	32	191 minidisk for IVPM1
94560-94815	256	11820-	32	191 minidisk for IVPM2
94816-122199	27384	11852-15274	3423	Spool space
122200-122711	512	15275-15338	64	Warm start data
122712-123223	512	15339-15402	64	I/O Error Recording area
123224-123735	512	15403-15466	64	Spool file checkpoint
123736-126015	2280	15467-15751	285	CP nucleus
126016-276015	150000	15752-36472	18750	Spool space
276016-276927	912	36471-45856	114	191 minidisk for EREP
366856-367311	456	45857-45913	57	191 minidisk for OPERATNS
367312-374607	7296	45914-46825	912	193 minidisk for OPERATNS
374608-380327	5720	46826-47540	715	xxx minidisk space for user option
380328-415327	35000	47541-51915	4375	T-DISK space
415328-417607	2280	51916-52200	285	xxx minidisk space for user option
417608-438507	20900	52201-	**	19D minidisk for MAINT
438508-557999	119492		**	xxx minidisk space for user option

Figure 58. VMSRES Layout for a 3370 Starter System

* The CMS nucleus must start on a 256 block boundary. 43520 displacement in the minidisk will contain the CMS nucleus.
The CMSL nucleus must start on a 256 block boundary. 44544 displacement in the minidisk will contain the CMSL nucleus.

** Not on a page boundary.

Appendix O. 3375 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3375 Directory Supplied with the Product Tape

Sample Directory for a 3375 Device

```
***************************************************************
*  3375    SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 3375 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$ NOLOG
 MDISK A01 3375 000 001 VMSRES R
 MDISK C01 3375 000 001 VMPK01 R
 MDISK D01 3375 000 001 VMSTGE R
 MDISK E01 3375 000 001 PROFPK R
 MDISK F01 3375 000 001 SQLPK  R
*
USER $TEMP$ NOLOG
 MDISK A04 3375 288 191 VMSRES R
 MDISK A09 3375 481 075 VMSRES R
 MDISK C03 3375 227 100 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK A10 3375 148 140 VMSRES R
*
```

Figure 59 (Part 1 of 13). Sample Directory for a 3375 Device

```
USER $CPNUC$ NOLOG
 MDISK A09 3375 953 006 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A05 3375 144 004 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 3375 001 014 VMSRES R
 MDISK B02 3375 001 018 VMPK01 R
 MDISK A04 3375 001 003 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A06 3375 949 002 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A06 3375 479 002 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 3375 947 002 VMSRES R
*
****************************************************************
*             SYSTEM RELATED USERIDS
****************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 204 003 VMPK01 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 147 001 VMPK01 MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 3375 221 005 VMPK01 MR RBATCH WBATCH MBATCH
*
```

Figure 59 (Part 2 of 13). Sample Directory for a 3375 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 019 003 VMPK01 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 3375 207 001 VMPK01 MR RCPRM WCPRM
 MDISK 192 3375 208 011 VMPK01 MR ALL WCPRM
 MDISK 291 3375 219 002 VMPK01 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 519 007 VMPK01 MR RCSP WCSP MCSP
 MDISK 193 3375 526 010 VMPK01 MR RCSP WCSP MCSP
 MDISK 502 3375 536 032 VMPK01 MR RCSP WCSP MCSP
 MDISK 503 3375 568 032 VMPK01 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 3375 197 004 VMPK01 M RMOVR WMOVR MMOVR
*
```

Figure 59 (Part 3 of 13). Sample Directory for a 3375 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3375 022 006 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 193 3375 036 009 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 195 3375 136 010 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 123 3375 000 959 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 148 001 VMPK01 MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 3375 935 002 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 3375 496 002 VMPK01 MR RFSF1 WFSF1
*
```

Figure 59 (Part 4 of 13). Sample Directory for a 3375 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 3375 470 011 VMPK01 MR RFSF WFSF
 MDISK 192 3375 481 004 VMPK01 MR RFSF WFSF
 MDISK 193 3375 485 004 VMPK01 MR RFSF WFSF
 MDISK 194 3375 489 002 VMPK01 MR RFSF WFSF
 MDISK 195 3375 491 002 VMPK01 MR RFSF WFSF
 MDISK 197 3375 493 001 VMPK01 MR RFSF WFSF
 MDISK 200 3375 494 001 VMPK01 MR RFSF WFSF
 MDISK 400 3375 495 001 VMPK01 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 045 030 VMPK01 MR ALL WIIPS MIIPS
 MDISK 193 3375 075 022 VMPK01 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 3375 146 001 VMPK01 MR RPSR WPSR MPSR
*
```

Figure 59 (Part 5 of 13). Sample Directory for a 3375 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 3375 133 003 VMPK01 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 028 003 VMPK01 MR RISPF WISPF MISPF
 MDISK 192 3375 097 025 VMPK01 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M   16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3375  951 001 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M 4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3375 952 001 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 59 (Part 6 of 13). Sample Directory for a 3375 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 3375 000 959 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3375 000 959 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3375 000 959 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 128 3375 000 959 PROFPK MW RSYSRES WSYSRES MSYSRES
 MDISK 129 3375 000 959 SQLPK  MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3375 070 074 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 3375 015 009 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 3375 024 032 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 3375 889 021 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D 3375 696 042 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E 3375 634 042 VMSRES MW ALL     WMAINT MMAINT
 MDISK 201 3375 362 032 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 293 3375 327 018 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3375 345 017 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 3375 912 021 VMSRES MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 3375 154 007 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 310 3375 442 025 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 319 3375 569 065 VMSRES MW ALL     WMAINT MMAINT
 MDISK 31A 3375 151 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 324 3375 498 021 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 325 3375 182 015 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 390 3375 201 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 393 3375 001 106 VMSTGE WR RMAINT  WMAINT
 MDISK 394 3375 107 120 VMSTGE WR RMAINT  WMAINT
 MDISK 396 3375 227 049 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 3375 170 012 VMPK01 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF 3375 000 959 CEPACK MR READ WRITE
*
```

Figure 59 (Part 7 of 13). Sample Directory for a 3375 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 132 001 VMPK01 MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 556 001 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 3375 557 012 VMSRES MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 910 002 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT  250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3375 432 005 PROFPK MR RDBM WDBM MDBM
 MDISK 161 3375 437 014 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD 3375 485 017 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE 3375 468 017 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF 3375 451 017 PROFPK MR RDBM WDBM MDBM
*
```

Figure 59 (Part 8 of 13). Sample Directory for a 3375 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT 250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 3375 378 011 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 3375 389 005 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 122 010 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 3375 XXX 002 XXXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 191 3375 165 005 VMPK01 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3375 373 005 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 59 (Part 9 of 13). Sample Directory for a 3375 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3375 394 003 PROFPK MR RCAL WCAL MCAL
 MDISK 196 3375 397 013 PROFPK MR RCAL WCAL MCAL
*
************************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
************************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3375 394 011 VMPK01 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3375 001 017 SQLPK W
 MDISK 193 3375 018 040 SQLPK R  RSQL WSQL
 MDISK 195 3375 058 021 SQLPK RR RSQL WSQL MSQL
 MDISK 200 3375 079 047 SQLPK R  RSQL WSQL
 MDISK 201 3375 126 013 SQLPK R  RSQL WSQL
 MDISK 202 3375 139 125 SQLPK R  RSQL WSQL
*
USER SQLUSER SQLUSER 2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 3375 264 004 SQLPK W
*
```

Figure 59 (Part 10 of 13). Sample Directory for a 3375 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT 250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 3375 304 007 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 3375 311 017 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 3375 355 018 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 3375 410 022 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 161 004 VMPK01 MR RDUMP WDUMP MDUMP
 MDISK 123 3375 000 959 VMSRES RR
 MDISK 125 3375 000 959 VMPK01 RR
 MDISK 126 3375 000 959 VMSTGE RR
 MDISK 128 3375 000 959 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 031 005 VMPK01 MR RVMAP WVMAP MVMAP
*
```

Figure 59 (Part 11 of 13). Sample Directory for a 3375 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3375 405 011 VMPK01 W RBATCH WBATCH
 MDISK 192 3375 416 014 VMPK01 W RBATCH WBATCH
 MDISK 193 3375 430 011 VMPK01 W RBATCH WBATCH
 MDISK 194 3375 441 001 VMPK01 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 3375 467 003 VMPK01 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3375 149 002 VMPK01 MR RUTIL WUTIL MUTIL
*
```

Figure 59 (Part 12 of 13). Sample Directory for a 3375 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3375 SYSTEM
*MDISK 150 3375 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3375 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3375 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3375 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3375 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3375 SYSTEM
*MDISK 350 3375 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3375 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3375 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3375 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 00000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 00000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 00000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 59 (Part 13 of 13). Sample Directory for a 3375 Device

Map of a 3375 Directory Sample

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	3375	OLTSEP	5FF	MR	000	958	959	
PROFPK	3375	MAINT	128	MW	000	958	959	
		SYSDUMP1	128	RR	000	958	959	
		$ALLOC$	E01	R	000	000	001	
		$SAVSYS$	A04	R	001	003	003	
					004	303	300	GAP
		SYSADMIN	191	MR	304	310	007	
		SYSADMIN	298	MR	311	327	017	
					328	354	027	GAP
		SYSADMIN	398	MR	355	372	018	
		SFBATCH	191	MR	373	377	005	
		PROMAIL	191	MR	378	388	011	
		PROMAIL	151	MR	389	393	005	
		SFCAL	191	MR	394	396	003	
		SFCAL	196	MR	397	409	013	
		SYSADMIN	399	MR	410	431	022	
		PRODBM	191	MR	432	436	005	
		PRODBM	161	MR	437	450	014	
		PRODBM	5FF	MR	451	467	017	
		PRODBM	5FE	MR	468	484	017	
		PRODBM	5FD	MR	485	501	017	
					502	958	457	GAP
SQLPK	3375	MAINT	129	MW	000	958	959	
		$ALLOC$	F01	R	000	000	001	
		SQLDBA	191	W	001	017	017	
		SQLDBA	193	R	018	057	040	
		SQLDBA	195	RR	058	078	021	
		SQLDBA	200	R	079	125	047	
		SQLDBA	201	R	126	138	013	
		SQLDBA	202	R	139	263	125	
		SQLUSER	191	W	264	267	004	
					268	958	691	GAP
VMPK01	3375	MAINT	125	MW	000	958	959	
		SYSDUMP1	125	RR	000	958	959	
		$ALLOC$	C01	R	000	000	001	
		$SAVSYS$	B02	R	001	018	018	
		CMSUSER	191	MR	019	021	003	
		DIRMAINT	191	MR	022	027	006	
		ISPVM	191	MR	028	030	003	
		VMAP	191	MR	031	035	005	
		DIRMAINT	193	MR	036	044	009	
		IIPS	191	MR	045	074	030	
		IIPS	193	MR	075	096	022	
		ISPVM	192	MR	097	121	025	
		PVM	191	MR	122	131	010	
		OP1	191	MR	132	132	001	
		ISMAINT	191	MR	133	135	003	
		DIRMAINT	195	MR	136	145	010	
		IPCS	191	MR	146	146	001	
		AUTOLOG1	191	MR	147	147	001	

Figure 60 (Part 1 of 3). Directory Map Sample for a 3375 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK01	3375	DISKACNT	191	MR	148	148	001	
		VMUTIL	191	MR	149	150	002	
		MAINT	31A	MW	151	153	003	
		MAINT	300	MW	154	160	007	
		SYSDUMP1	191	MR	161	164	004	
		RSCS	191	MR	165	169	005	
		MAINT	3A0	MW	170	181	012	
		MAINT	325	MW	182	196	015	
		DATAMOVR	191	M	197	200	004	
		MAINT	390	MW	201	203	003	
		ADMIN	191	MR	204	206	003	
		CPRM	191	MR	207	207	001	
		CPRM	192	MR	208	218	011	
		CPRM	291	MR	219	220	002	
		CMSBATCH	195	MR	221	225	005	
					226	226	001	GAP
		$TEMP$	C03	R	227	326	100	
		MAINT	293	MW	327	344	018	
		MAINT	294	MW	345	361	017	
		MAINT	201	MW	362	393	032	
		SMART	191	MR	394	404	011	
		VMBATCH	191	W	405	415	011	
		VMBATCH	192	W	416	429	014	
		VMBATCH	193	W	430	440	011	
		VMBATCH	194	W	441	441	001	
		MAINT	310	MW	442	466	025	
		VMBATCH1	191	W	467	469	003	
		FSFCNTRL	191	MR	470	480	011	
		FSFCNTRL	192	MR	481	484	004	
		FSFCNTRL	193	MR	485	488	004	
		FSFCNTRL	194	MR	489	490	002	
		FSFCNTRL	195	MR	491	492	002	
		FSFCNTRL	197	MR	493	493	001	
		FSFCNTRL	200	MR	494	494	001	
		FSFCNTRL	400	MR	495	495	001	
		FSFADMIN	192	MR	496	497	002	
		MAINT	324	MW	498	518	021	
		CSPUSER	191	MR	519	525	007	
		CSPUSER	193	MR	526	535	010	
		CSPUSER	502	MR	536	567	032	
		CSPUSER	503	MR	568	599	032	
					600	958	359	GAP
VMSRES	3375	DIRMAINT	123	MW	000	958	959	
		MAINT	123	MW	000	958	959	
		SYSDUMP1	123	RR	000	958	959	
		$ALLOC$	A01	R	000	000	001	
		$SAVSYS$	A02	R	001	014	014	
		MAINT	191	MW	015	023	009	
		MAINT	194	MW	024	055	032	
					056	069	014	GAP
		MAINT	190	MW	070	143	074	
		$DIRECT$	A05	R	144	147	004	
		$TDISK$	A10	R	148	287	140	
		$TEMP$	A04	R	288	478	191	

Figure 60 (Part 2 of 3). Directory Map Sample for a 3375 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSRES	3375	$SYSCKP$	A06	R	479	480	002	
		$TEMP$	A09	R	481	555	075	
		OPERATNS	191	MR	556	556	001	
		OPERATNS	193	MR	557	568	012	
		MAINT	319	MW	569	633	065	
		MAINT	19E	MW	634	675	042	
					676	695	020	GAP
		MAINT	19D	MW	696	737	042	
					738	888	151	GAP
		MAINT	196	MW	889	909	021	
		OPERATOR	191	MR	910	911	002	
		MAINT	296	MW	912	932	021	
					933	934	002	GAP
		EREP	191	WR	935	936	002	
					937	946	010	GAP
		$SYSWRM$	A07	R	947	948	002	
		$SYSERR$	A06	R	949	950	002	
		IVPM1	191	WR	951	951	001	
		IVPM2	191	WR	952	952	001	
		$CPNUC$	A09	R	953	958	006	
VMSTGE	3375	MAINT	126	MW	000	958	959	
		SYSDUMP1	126	RR	000	958	959	
		$ALLOC$	D01	R	000	000	001	
		MAINT	393	WR	001	106	106	
		MAINT	394	WR	107	226	120	
		MAINT	396	WR	227	275	049	
					276	958	683	GAP

Figure 60 (Part 3 of 3). Directory Map Sample for a 3375 Device

3375 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3375   SP STARTER SYSTEM SAMPLE      11/30/81
         COPY  OPTIONS
DMKRIO   CSECT
         PRINT NOGEN
         SPACE 3
*
**********************************************************************
*  THIS A SAMPLE DMKRIO ASSEMBLE FOR THE 3375 -  THIS SAMPLE IS TO
*  BE USED AS A GUIDELINE IN PREPARING YOUR OWN DMKRIO FILE OR IT
*  MAY BE EDITED FOR USE BY ALTERING AND DELETING UNUSED ENTRIES.
**********************************************************************
*
*
**********************************************************************
*
*      C L U S T E R   A N D   L I N E   C O N T R O L L E R S
*
**********************************************************************
*
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
*
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 61 (Part 1 of 5). Sample DMKRIO File for a 3375 Device

```
*
**********************************************************************
*
*                     C H A N N E L   Z E R O
*
*    NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*          016 MAY BE USED AS DISPLAY CONSOLE FOR 168 OR 165-II.
*
**********************************************************************
*
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=S
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
SPECONS  RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
RDR      RDEVICE ADDRESS=01C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=01D,DEVTYPE=3525,CLASS=G
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
GRAF     RDEVICE ADDRESS=(020,32),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
*
**********************************************************************
*
*                     C H A N N E L   O N E
*
**********************************************************************
*
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,32),DEVTYPE=3277
         RDEVICE ADDRESS=(140,16),DEVTYPE=3375
         RDEVICE ADDRESS=(150,16),DEVTYPE=3375
         RDEVICE ADDRESS=(160,16),DEVTYPE=3375
         RDEVICE ADDRESS=(170,16),DEVTYPE=3375
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1C0,16),DEVTYPE=3375
         EJECT
```

Figure 61 (Part 2 of 5). Sample DMKRIO File for a 3375 Device

```
*
**********************************************************************
*
*                     C H A N N E L    T W O
*
**********************************************************************
*
         RDEVICE ADDRESS=(220,32),DEVTYPE=3277
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3375
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         EJECT
*
**********************************************************************
*
*                     C H A N N E L    T H R E E
*
**********************************************************************
*
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3375
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         EJECT
*
**********************************************************************
*
*                     C H A N N E L    F O U R
*
**********************************************************************
*
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3375
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
*
**********************************************************************
*
*                     C H A N N E L    F I V E
*
**********************************************************************
*
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
```

Figure 61 (Part 3 of 5). Sample DMKRIO File for a 3375 Device

```
*
**********************************************************************
*
*                     C O N T R O L    U N I T S
*
**********************************************************************
*
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=020,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=32-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=160,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU2   RCTLUNIT ADDRESS=220,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=580,CUTYPE=3803,FEATURE=32-DEVICE
         EJECT
*
**********************************************************************
*
*                         C H A N N E L S
*
**********************************************************************
*
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
*
```

Figure 61 (Part 4 of 5). Sample DMKRIO File for a 3375 Device

```
***********************************************************************
*
*                C O N S O L E
*
***********************************************************************
*
         RIOGEN CONS=014,ALTCONS=(01F,009,01A,016,010)
         END
***********************************************************************
*
*                              E N D
*
***********************************************************************
```

Figure 61 (Part 5 of 5). Sample DMKRIO File for a 3375 Device

3375 DMKSYS File Supplied With The Product Tape

The 3375 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=3375,                                           X
               SYSCLR=YES,                                             X
               SYSNUC=953,                                             X
               SYSWRM=(947,2),                                         X
               SYSERR=(949,2),                                         X
               SYSCKP=(479,2)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO,                                                  X
               MP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 62. Sample DMKSYS File for a 3375 Device

3375 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT       VM/SP REL 3       3375 SAMPLE'
         SPACE
*
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSCYL=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSCYL  - THE CYLINDER ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 63 (Part 1 of 8). Sample DMKSNT File for a 3375 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSCYL=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*
         EJECT
*
*********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
*********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*
*     HEX LOAD ADDRESS FOR SEGMENT 25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*     ( THE ALLOCATIONS ARE BASED ON 96 PAGES/3375 CYLINDER )
*          CYL 1, PAGE 1 TO CYL 2, PAGE 42 (138 PAGES)
*         137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*
*********************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(001,1),                                        X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=70,                                              X
               VSYSRES=VMSRES
         EJECT
```

Figure 63 (Part 2 of 8). Sample DMKSNT File for a 3375 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 2, PAGE 43 TO CYL 3, PAGE 84 (138 PAGES)
*********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(002,43),                                       X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=70,                                              X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 85 TO CYL 4, PAGE 05 ( 17 PAGES )
*********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(003,85),                                       X
               SYSPGNM=(784-799),                                      X
               SYSPGCT=16,                                             X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 4, PAGE 06 TO CYL 4, PAGE 54 (49 PAGES)
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(004,06),                                       X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 63 (Part 3 of 8). Sample DMKSNT File for a 3375 Device

```
*******************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 4, PAGE 55 TO CYL 5, PAGE 71 ( 113 PAGES )
********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(004,55),                                       X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*******************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 5, PAGE 72 TO CYL 07, PAGE 24 (145 PAGES)
********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(005,72),                                       X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*******************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 7, PAGE 25 TO CYL 7, PAGE 41 (17 PAGES)
*******************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(007,25),                                       X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 63 (Part 4 of 8). Sample DMKSNT File for a 3375 Device

```
*********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 7, PAGE 42 TO CYL 7, PAGE 58 (17 PAGES)
*********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(007,042),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
*********************************************************************
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 7, PAGE 59 TO CYL 7, PAGE 75 (17 PAGES)
*********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(007,059),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
*********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 07, PAGE 76 TO CYL 08, PAGE 76 (97 PAGES)
*********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                      X
               SYSSIZE=384K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(007,076),                                      X
               SYSPGCT=96,                                             X
               SYSPGNM=(1072-1167),                                    X
               SYSHRSG=(67,68,69,70,71,72),                            X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 63 (Part 5 of 8). Sample DMKSNT File for a 3375 Device

```
***********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 08, PAGE 77 TO CYL  08 PAGE 93 (17 PAGES)
***********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(08,077),                                       X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
***********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 08, PAGE 94 TO CYL 09, PAGE 94 (97 PAGES)
***********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(08,94),                                        X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
***********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 09, PAGE 95 TO CYL 10, PAGE 95 (97 PAGES)
***********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(09,95),                                        X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 63 (Part 6 of 8). Sample DMKSNT File for a 3375 Device

```
*********************************************************************
*  IIPS        (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 10, PAGE 96 TO CYL 11, PAGE 64 (65 PAGES)
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(010,96),                                       X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE 1  (97 PAGES)
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(001,01),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 5, PAGE 5 TO CYL 6, PAGE 5 (97 PAGES)
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(005,05),SYSPGCT=96,              X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 6, PAGE 6 TO CYL 6, PAGE 54 (49 PAGES)
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(006,006),                        X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 63 (Part 7 of 8). Sample DMKSNT File for a 3375 Device

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 6, PAGE 55 TO CYL 9, PAGE 07 (241 PAGES)
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(06,55),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 09, PAGE 08 TO CYL 13, PAGE 72 (449 PAGES)
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(09,08),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 3, PAGE 1 (193 PAGES)
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 63 (Part 8 of 8). Sample DMKSNT File for a 3375 Device

VMSRES Pack Layout for 3375 Starter System

Real Cylinder Address	Number of Cylinders'	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-14	4	Not used
15-23	9	191 minidisk for MAINT
24-55	32	194 minidisk for MAINT (CP)
56-69	14	xxx minidisk space for user option
70-143	74	190 minidisk for MAINT (CMS system disk)*
144-147	4	VM/SP directory
148-287	140	T-DISK space
288-478	191	Spool space
479-480	2	Spool file checkpoint
481-555	75	Spool space
556	1	191 minidisk for OPERATNS
557-568	12	193 minidisk for OPERATNS
569-633	65	319 minidisk for MAINT
634-675	42	19E minidisk for MAINT
676-695	20	xxx minidisk for user option
696-737	42	19D minidisk for MAINT (HELP)
738-909	172	xxx minidisk for user option
910-911	2	191 minidisk for OPERATOR
912-934	23	xxx minidisk for user option
935-936	2	191 minidisk for EREP
937-946	10	xxx minidisk for user option
947-948	2	Warm start data
949-950	2	I/O Error Recording area
951	1	191 minidisk for IVPM1
952	1	191 minidisk for IVPM2
953-958	6	CP nucleus

Figure 64. VMSRES Layout for a 3375 Starter System

* Specify cylinder 70 to load the CMS nucleus which occupies the first 2 cylinders of the recompiled area.
Specify cylinder 72 to load the CMSL nucleus which occupies the last 2 cylinders of the recompiled area.

Appendix P. 3380 Based System Sample Files

These directories are intended to be used as *samples only.* They show the minidisks required and the minidisk sizes for each userid.

After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

3380 Directory Supplied with the Product Tape

Sample Directory for a 3380 Device

```
***************************************************************
*  3380    SYSTEM DIRECTORY
*  THE ADDRESSES 123, 124, AND 125 ARE VIRTUAL ADDRESSES.
*  THE ADDRESS 123 IS CRITICAL SINCE IT USED IN DMKSYS,
*  THE DIRECTORY, AND THE SERVICE ENVIRONMENTS OF THE
*  INTERACTIVE PRODUCTIVITY FACILITY. DO NOT CHANGE THIS
*  ADDRESS. IF YOU STILL WANT TO CHANGE IT, REMEMBER IT MUST
*  BE CHANGED IN DMKSYS, ALL SERVICE ENVIRONMENTS, THE
*  'DIRECTORY' STATEMENT BELOW, AND IN THE 'MDISK'
*  STATEMENTS FOUND UNDER THE USERID  'MAINT'.
*  NOTE: REMEMBER THESE ARE ONLY VIRTUAL ADDRESSES NOT REAL
*  ADDRESSES, SO THERE IS NO NEED TO CHANGE THEM TO MATCH
*  YOUR HARDWARE ADDRESSES.
*  FURTHER INFORMATION IS CONTAINED IN THE SYSTEM
*  INSTALLATION GUIDE.
***************************************************************
*
DIRECTORY 123 3380 VMSRES
*
***************************************************************
*         SYSTEM RESERVED AREAS NOT FOR MINIDISKS
***************************************************************
*
USER $ALLOC$ NOLOG
 MDISK A01 3380 000 001 VMSRES R
 MDISK C01 3380 000 001 VMPK01 R
 MDISK D01 3380 000 001 VMSTGE R
 MDISK E01 3380 000 001 PROFPK R
 MDISK F01 3380 000 001 SQLPK  R
*
USER $TEMP$ NOLOG
 MDISK A04 3380 402 040 VMSRES R
 MDISK A09 3380 443 040 VMSRES R
 MDISK C03 3380 327 100 VMPK01 R
*
USER $TDISK$ NOLOG
 MDISK A10 3380 483 100 VMSRES R
*
```

Figure 65 (Part 1 of 13). Sample Directory for a 3380 Device

```
USER $CPNUC$ NOLOG
 MDISK A09 3380 881 004 VMSRES R
*
USER $DIRECT$ NOLOG
 MDISK A05 3380 097 002 VMSRES R
*
USER $SAVSYS$ NOLOG
 MDISK A02 3380 001 010 VMSRES R
 MDISK B02 3380 001 012 VMPK01 R
 MDISK A04 3380 001 002 PROFPK R
*
USER $SYSERR$ NOLOG
 MDISK A06 3380 879 002 VMSRES R
*
USER $SYSCKP$ NOLOG
 MDISK A06 3380 442 001 VMSRES R
*
USER $SYSWRM$ NOLOG
 MDISK A07 3380 877 002 VMSRES R
*
**************************************************************
*             SYSTEM RELATED USERIDS
**************************************************************
*
USER ADMIN ADMIN 1664K 16M ABCDEFG
 ACCOUNT 1 SYSADMIN
 IPL CMS PARM AUTOCR
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 166 001 VMPK01 MR RADMIN WADMIN
*
USER AUTOLOG1 AUTOLOG 512K 1M ABCDEG
 ACCOUNT 2 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 106 001 VMPK01 MR RAUTOLOG WAUTOLOG MAUTOLOG
*
USER CMSBATCH BATCH 1M 2M G
 ACCOUNT 3 SYSTEM
 OPTION ACCT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 MDISK 195 3380 129 001 VMPK01 MR RBATCH WBATCH MBATCH
*
```

Figure 65 (Part 2 of 13). Sample Directory for a 3380 Device

```
USER CMSUSER CMSUSER 1M 3M G
 ACCOUNT 101 USER01
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 013 003 VMPK01 MR RCMS WCMS MCMS
*
USER CPRM CPRM 512K 2M G
 ACCOUNT 4 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK OPERATNS 193 193 RR
 MDISK 191 3380 167 001 VMPK01 MR RCPRM WCPRM
 MDISK 192 3380 168 007 VMPK01 MR ALL WCPRM
 MDISK 291 3380 175 001 VMPK01 MR RCPRM WCPRM
*
USER CSPUSER CSPUSER 1M 3M G
 ACCOUNT 101
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 243 004 VMPK01 MR RCSP WCSP MCSP
 MDISK 193 3380 247 007 VMPK01 MR RCSP WCSP MCSP
 MDISK 502 3380 254 020 VMPK01 MR RCSP WCSP MCSP
 MDISK 503 3380 274 020 VMPK01 MR RCSP WCSP MCSP
*
USER DATAMOVR MOVR 512K 1M G
 ACCOUNT 5 SYSADMIN
 OPTION ACCT ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 319 319 RR
 LINK DIRMAINT 191 193 RR
 MDISK 191 3380 142 003 VMPK01 M RMOVR WMOVR MMOVR
*
```

Figure 65 (Part 3 of 13). Sample Directory for a 3380 Device

```
USER DIRMAINT DIRM 1M 2M BG
 ACCOUNT 6 SYSADMIN
 OPTION REALTIMER ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 SPECIAL OFF TIMER
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3380 130 003 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 193 3380 087 009 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 195 3380 133 009 VMPK01 MR RDIRM WDIRM MDIRM
 MDISK 123 3380 000 885 VMSRES MW
*
USER DISKACNT ACNT 512K 1M G
 ACCOUNT 7 ACCNTNG
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 107 001 VMPK01 MR RACNT WACNT MACNT
*
USER EREP IBMCE 768K 2M FG
 ACCOUNT EREP IBMCE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 201 192 RR
 MDISK 191 3380 099 001 VMSRES WR READ WRITE
*
USER FSFADMIN FSFADMIN 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 192 3380 229 001 VMPK01 MR RFSF1 WFSF1
*
```

Figure 65 (Part 4 of 13). Sample Directory for a 3380 Device

```
USER FSFCNTRL FSFCNTRL 2M 16M BG
 ACCOUNT
 OPTION ECMODE BMX MAXCONN 256
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK FSFADMIN 192 198 RR
 MDISK 191 3380 209 007 VMPK01 MR RFSF WFSF
 MDISK 192 3380 216 003 VMPK01 MR RFSF WFSF
 MDISK 193 3380 219 003 VMPK01 MR RFSF WFSF
 MDISK 194 3380 222 002 VMPK01 MR RFSF WFSF
 MDISK 195 3380 224 002 VMPK01 MR RFSF WFSF
 MDISK 197 3380 226 001 VMPK01 MR RFSF WFSF
 MDISK 200 3380 227 001 VMPK01 MR RFSF WFSF
 MDISK 400 3380 228 001 VMPK01 MR RFSF WFSF
*
USER FSFTASK1 FSFTASK1 512K 1M G
 ACCOUNT
 OPTION BMX MAXCONN 2
 IUCV ALLOW PRIORITY MSGLIMIT 255
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK FSFCNTRL 191 191 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*
USER IIPS IIPS 2M 2M G
 ACCOUNT 8 INSTR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 044 020 VMPK01 MR ALL WIIPS MIIPS
 MDISK 193 3380 064 015 VMPK01 MR ALL WIIPS MIIPS
*
USER IPCS PSR 512K 1M BCEG
 ACCOUNT 9 CE-ROOM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 LINK MAINT 319 319 RR
 LINK MAINT 31A 31A RR
 LINK OPERATNS 193 192 RR
 MDISK 191 3380 105 001 VMPK01 MR RPSR WPSR MPSR
*
```

Figure 65 (Part 5 of 13). Sample Directory for a 3380 Device

```
USER ISMAINT ISMAINT 1200K 2100K G
 ACCOUNT 105 ISMAINT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 W
 MDISK 191 3380 016 002 VMPK01 MR RISM WISM MISM
*
USER ISPVM ISPVM 1M 10M BEG
 ACCOUNT 104 USER04
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 018 003 VMPK01 MR RISPF WISPF MISPF
 MDISK 192 3380 021 018 VMPK01 MR RISPF WISPF MISPF
*
USER IVPM1 IVPASS 3M   16M G
 ACCOUNT ACT4 IVPM1
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3380  875 001 VMSRES WR READ WRITE
*
USER IVPM2 IVPASS 3M   4M G
 ACCOUNT ACT5 IVPM2
 CONSOLE 009 3210
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 194 194 RR
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3380 876 001 VMSRES WR READ WRITE
*
USER LEV2VM LEV2VM 4M 8M BCDEFG 64 |
 ACCOUNT 400 SYSPROG
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403  A
 SPECIAL 120 3270
*
```

Figure 65 (Part 6 of 13). Sample Directory for a 3380 Device

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK DIRMAINT 191 197 W
 LINK RSCS     191 202 MW
 LINK OPERATNS 193 491 W
 LINK IIPS     191 492 W
 LINK IIPS     193 493 W
 LINK VMAP     191 494 W
 LINK ISMAINT  191 495 W
 LINK PVM      191 496 W
 LINK ISPVM    192 497 W
 LINK CPRM     191 498 W
 LINK CPRM     291 499 W
 MDISK 123 3380 000 885 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3380 000 885 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3380 000 885 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 128 3380 000 885 PROFPK MW RSYSRES WSYSRES MSYSRES
 MDISK 129 3380 000 885 SQLPK  MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3380 052 045 VMSRES MW ALL     WMAINT MMAINT
 MDISK 191 3380 011 008 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 194 3380 019 021 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 196 3380 172 014 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 19D 3380 221 027 VMSRES MW ALL     WMAINT MMAINT
 MDISK 19E 3380 100 027 VMSRES MW ALL     WMAINT MMAINT
 MDISK 201 3380 832 021 VMSRES MW RMAINT  WMAINT MMAINT
 MDISK 293 3380 853 011 VMSRES MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3380 864 011 VMSRES MW RCPAUX  WCPAUX  MCPAUX
 MDISK 296 3380 248 014 VMSRES MW RCPAUX  WCPAUX  MCPAUX
*
 MDISK 300 3380 145 005 VMPK01 MW RMAINT  WMAINT MMAINT
 MDISK 310 3380 150 016 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 319 3380 127 045 VMSRES MW ALL     WMAINT MMAINT
 MDISK 31A 3380 111 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 324 3380 230 013 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 325 3380 114 009 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 390 3380 123 003 VMPK01 MW ALL     WMAINT MMAINT
 MDISK 393 3380 001 068 VMSTGE WR RMAINT  WMAINT
 MDISK 394 3380 809 076 VMSTGE WR RMAINT  WMAINT
 MDISK 396 3380 069 031 VMSTGE WR RMAINT  WMAINT
 MDISK 3A0 3380 079 008 VMPK01 MW ALL     WMAINT MMAINT
*
USER OLTSEP IBMCE 1M 1M FG
 ACCOUNT OLTSEP IBMCE
 OPTION REALTIMER ECMODE
 IPL 5FF
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 MDISK 5FF 3380 000 885 CEPACK MR READ WRITE
*
```

Figure 65 (Part 7 of 13). Sample Directory for a 3380 Device

```
USER OP1 OP1 1M 4M ABCDEFG
 ACCOUNT 12 OPERATOR
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 110 001 VMPK01 MR ROP1 WOP1 MOP1
*
USER OPERATNS IPCS 1M 2M BCEG
 ACCOUNT 13 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 212 001 VMSRES MR RIPCS WIPCS MIPCS
 MDISK 193 3380 213 008 VMSRES MR RIPCS WIPCS MIPCS
*
USER OPERATOR OPERATOR  3M 16M ABCDEFG
 ACCOUNT 2 OPERATOR
 CONSOLE 009 3215 T MAINT
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 040 001 VMSRES MR ROPER WOPER MOPER
*
USER OSVS1 OSVS1 8M 16M G
 ACCOUNT 300 OSVS1
 OPTION ECMODE REALTIMER BMX
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
*
USER PRODBM PRODBM 1M 2M G
 ACCOUNT  250 PRODBM
 OPTION REALTIMER
 IPL CMS
 CONSOLE  009  3215
 SPOOL  C  2540  READER *
 SPOOL  D  2540  PUNCH A
 SPOOL  E  1403  A
 LINK  MAINT 19D  19D  RR
 LINK  MAINT 19E  19E  RR
 LINK  MAINT 190  190  RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3380 434 003 PROFPK MR RDBM WDBM MDBM
 MDISK 161 3380 437 009 PROFPK MR RDBM WDBM MDBM
 MDISK 5FD 3380 468 011 PROFPK MR RDBM WDBM MDBM
 MDISK 5FE 3380 457 011 PROFPK MR RDBM WDBM MDBM
 MDISK 5FF 3380 446 011 PROFPK MR RDBM WDBM MDBM
*
```

Figure 65 (Part 8 of 13). Sample Directory for a 3380 Device

```
USER PROMAIL PROMAIL 1M 2M G
 ACCOUNT  250 PROMAIL
 OPTION REALTIMER
 IPL CMS
 CONSOLE  009  3215
 SPOOL  C  2540  READER *
 SPOOL  D  2540  PUNCH A
 SPOOL  E  1403  A
 LINK  MAINT 19D  19D  RR
 LINK  MAINT 19E  19E  RR
 LINK  MAINT 190  190  RR
 LINK SYSADMIN 399 399 RR
 LINK PRODBM   191 395 RR
 MDISK 191 3380 400 007 PROFPK MR RMAIL WMAIL MMAIL
 MDISK 151 3380 407 003 PROFPK MR RMAIL WMAIL MMAIL
*
USER PVM PVM 1M 2M BG
 ACCOUNT 14 PASSTHRU
 OPTION ECMODE
 IPL CMS
 CONSOLE 01F 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 096 004 VMPK01 MR RPVM WPVM MPVM
*
USER ROUTER ROUTER 512K 2M G
 ACCOUNT 210 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
*MDISK 191 3380 XXX 002 XXXXXX MR VSEIPO VSEIPO VSEIPO
*
USER RSCS RSCS 512K 1M G
 ACCOUNT 15 SYSTEM
 IPL 191
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3380 100 005 VMPK01 MR RRSCS WRSCS MRSCS
*
USER SFBATCH SFBATCH 1M 2M G
 ACCOUNT 250 SFBATCH
 OPTION REALTIMER
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3380 397 003 PROFPK MR RSFBAT WSFBAT MSFBAT
*
```

Figure 65 (Part 9 of 13). Sample Directory for a 3380 Device

```
USER SFCAL SFCAL 1M 2M G
 ACCOUNT 250 SFCAL
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK SYSADMIN 399 399 RR
 MDISK 191 3380 410 002 PROFPK MR RCAL WCAL MCAL
 MDISK 196 3380 412 008 PROFPK MR RCAL WCAL MCAL
*
************************************************************************
* WARNING TO THE SYSTEM PROGRAMMER:
*   When changing the read password for SFCAL 196 make sure that the
*   same change is made in the file CALPARM AA1FILE on the minidisk
*   SFBATCH 191 (see PROFS PRPQ Install and Admin. Guide for details.)
************************************************************************
*
USER SMART SMART 1536K 2M CEG
 ACCOUNT
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 319 319 RR
 MDISK 191 3380 176 007 VMPK01 MR RSMART WSMART MSMART
*
USER SQLDBA SQLDBAPW 6M 6M G
 ACCOUNT
 OPTION REALTIMER MAXCONN 25
 IUCV ALLOW
 IPL CMS
 CONSOLE 009 3215 T OPERATOR
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 MDISK 191 3380 001 012 SQLPK W
 MDISK 193 3380 013 028 SQLPK R  RSQL WSQL
 MDISK 195 3380 041 015 SQLPK RR RSQL WSQL MSQL
 MDISK 200 3380 056 034 SQLPK R  RSQL WSQL
 MDISK 201 3380 090 008 SQLPK R  RSQL WSQL
 MDISK 202 3380 098 119 SQLPK R  RSQL WSQL
*
USER SQLUSER SQLUSER 2M 2M G
 ACCOUNT
 OPTION REALTIMER
 IUCV SQLDBA
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1443
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK SQLDBA 195 195 RR
 MDISK 191 3380 217 003 SQLPK W
*
```

Figure 65 (Part 10 of 13). Sample Directory for a 3380 Device

```
USER SYSADMIN SYSADMIN 1M 16M EG
 ACCOUNT 250 SYSADMIN
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 190 190 RR
 LINK PRODBM 191 4FA RR
 LINK PRODBM 161 161 RR
 LINK PRODBM 5FF 5FF RR
 LINK PRODBM 5FE 5FE RR
 LINK PRODBM 5FD 5FD RR
 MDISK 191 3380 353 004 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 298 3380 357 010 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 398 3380 386 011 PROFPK MR RADMIN WADMIN MADMIN
 MDISK 399 3380 420 014 PROFPK MR RADMIN WADMIN MADMIN
*
USER SYSDUMP1 SYSDUMP 1M 1M BG
 ACCOUNT 16 SYSTEM
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 126 003 VMPK01 MR RDUMP WDUMP MDUMP
 MDISK 123 3380 000 885 VMSRES RR
 MDISK 125 3380 000 885 VMPK01 RR
 MDISK 126 3380 000 885 VMSTGE RR
 MDISK 128 3380 000 885 PROFPK RR
*
USER VMAP VMAP 2M 4M BEG
 ACCOUNT 17 SYSPROG
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 039 005 VMPK01 MR RVMAP WVMAP MVMAP
*
```

Figure 65 (Part 11 of 13). Sample Directory for a 3380 Device

```
USER VMBATCH VMBATCH 1M 2M ABCDEFG
 ACCOUNT
 OPTION ACCT BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 MDISK 191 3380 183 007 VMPK01 W RBATCH WBATCH
 MDISK 192 3380 190 009 VMPK01 W RBATCH WBATCH
 MDISK 193 3380 199 007 VMPK01 W RBATCH WBATCH
 MDISK 194 3380 206 001 VMPK01 W RBATCH WBATCH
*
USER VMBATCH1 VMBATCH1 2M 2M G
 ACCOUNT
 OPTION BMX
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK VMBATCH 194 192 RR
 MDISK 191 3380 207 002 VMPK01 W RBATCH1 VMBATCH1
*
USER VMUTIL VMUTIL 512K 4M ABDEG
 ACCOUNT 18 SYSTEM
 OPTION ECMODE
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19D 19D RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
 MDISK 191 3380 108 002 VMPK01 MR RUTIL WUTIL MUTIL
*
```

Figure 65 (Part 12 of 13). Sample Directory for a 3380 Device

```
USER VSEIPO VSEIPO 16M 16M G
 ACCOUNT 203 DOSSYS
 IPL CMS
 OPTION ECMODE BMX REALTIMER
 CONSOLE 01F 3215
 SPECIAL 080 3270
 SPECIAL 081 3270
 SPECIAL 082 3270
 SPECIAL 083 3270
 SPECIAL 084 3270
 SPECIAL 085 3270
 SPECIAL 086 3270
 SPECIAL 087 3270
 SPECIAL 088 3270
 SPECIAL 089 3270
 SPECIAL 08A 3270
 SPECIAL 08B 3270
 SPOOL 00C 3505 A
 SPOOL 00D 3525 A
 SPOOL 00E 1403 A
 SPOOL 05D 3525 A
 SPOOL 05E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
* 3380 SYSTEM
*MDISK 150 3380 000 411 DOSRES MR VSEIPO  VSEIPO
*MDISK 151 3380 000 411 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 152 3380 000 411 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 153 3380 000 411 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 154 3380 000 411 SYSWK4 MR VSEIPO  VSEIPO
* 3340 SYSTEM
*MDISK 1C0 3340 000 698 DOSRES MR VSEIPO  VSEIPO
*MDISK 1C1 3340 000 698 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 1C2 3340 000 698 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 1C3 3340 000 698 SYSWK3 MR VSEIPO  VSEIPO
*MDISK 1C4 3340 000 698 SYSWK4 MR VSEIPO  VSEIPO
* 3380 SYSTEM
*MDISK 350 3380 000 560 DOSRES MR VSEIPO  VSEIPO
*MDISK 351 3380 000 560 SYSWK1 MR VSEIPO  VSEIPO
*MDISK 352 3380 000 560 SYSWK2 MR VSEIPO  VSEIPO
*MDISK 353 3380 000 560 SYSWK3 MR VSEIPO  VSEIPO
* 3370 SYSTEM
*MDISK 240 FB-512 00000 558000 DOSRES MR VSEIPO VSEIPO
*MDISK 241 FB-512 00000 558000 SYSWK1 MR VSEIPO VSEIPO
*MDISK 242 FB-512 00000 558000 SYSWK2 MR VSEIPO VSEIPO
*
USER VSEMAINT VSEMAINT 1M 4M BG
 ACCOUNT 211 DOSSYS
 IPL CMS
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH A
 SPOOL 00E 1403 A
 LINK MAINT 190 190 RR
 LINK MAINT 19E 19E RR
 LINK MAINT 300 300 RR
* LINK MAINT 301 301 RR
*
```

Figure 65 (Part 13 of 13). Sample Directory for a 3380 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
CEPACK	3380	OLTSEP	5FF	MR	000	884	885	
PROFPK	3380	MAINT	128	MW	000	884	885	
		SYSDUMP1	128	RR	000	884	885	
		$ALLOC$	E01	R	000	000	001	
		$SAVSYS$	A04	R	001	002	002	
					003	352	350	GAP
		SYSADMIN	191	MR	353	356	004	
		SYSADMIN	298	MR	357	366	010	
					367	385	019	GAP
		SYSADMIN	398	MR	386	396	011	
		SFBATCH	191	MR	397	399	003	
		PROMAIL	191	MR	400	406	007	
		PROMAIL	151	MR	407	409	003	
		SFCAL	191	MR	410	411	002	
		SFCAL	196	MR	412	419	008	
		SYSADMIN	399	MR	420	433	014	
		PRODBM	191	MR	434	436	003	
		PRODBM	161	MR	437	445	009	
		PRODBM	5FF	MR	446	456	011	
		PRODBM	5FE	MR	457	467	011	
		PRODBM	5FD	MR	468	478	011	
					479	884	406	GAP
SQLPK	3380	MAINT	129	MW	000	884	885	
		$ALLOC$	F01	R	000	000	001	
		SQLDBA	191	W	001	012	012	
		SQLDBA	193	R	013	040	028	
		SQLDBA	195	RR	041	055	015	
		SQLDBA	200	R	056	089	034	
		SQLDBA	201	R	090	097	008	
		SQLDBA	202	R	098	216	119	
		SQLUSER	191	W	217	219	003	
					220	884	665	GAP
VMPK01	3380	MAINT	125	MW	000	884	885	
		SYSDUMP1	125	RR	000	884	885	
		$ALLOC$	C01	R	000	000	001	
		$SAVSYS$	B02	R	001	012	012	
		CMSUSER	191	MR	013	015	003	
		ISMAINT	191	MR	016	017	002	
		ISPVM	191	MR	018	020	003	
		ISPVM	192	MR	021	038	018	
		VMAP	191	MR	039	043	005	
		IIPS	191	MR	044	063	020	
		IIPS	193	MR	064	078	015	
		MAINT	3A0	MW	079	086	008	
		DIRMAINT	193	MR	087	095	009	
		PVM	191	MR	096	099	004	
		RSCS	191	MR	100	104	005	
		IPCS	191	MR	105	105	001	
		AUTOLOG1	191	MR	106	106	001	
		DISKACNT	191	MR	107	107	001	

Figure 66 (Part 1 of 3). Directory Map Sample for a 3380 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMPK01	3380	VMUTIL	191	MR	108	109	002	
		OP1	191	MR	110	110	001	
		MAINT	31A	MW	111	113	003	
		MAINT	325	MW	114	122	009	
		MAINT	390	MW	123	125	003	
		SYSDUMP1	191	MR	126	128	003	
		CMSBATCH	195	MR	129	129	001	
		DIRMAINT	191	MR	130	132	003	
		DIRMAINT	195	MR	133	141	009	
		DATAMOVR	191	M	142	144	003	
		MAINT	300	MW	145	149	005	
		MAINT	310	MW	150	165	016	
		ADMIN	191	MR	166	166	001	
		CPRM	191	MR	167	167	001	
		CPRM	192	MR	168	174	007	
		CPRM	291	MR	175	175	001	
		SMART	191	MR	176	182	007	
		VMBATCH	191	W	183	189	007	
		VMBATCH	192	W	190	198	009	
		VMBATCH	193	W	199	205	007	
		VMBATCH	194	W	206	206	001	
		VMBATCH1	191	W	207	208	002	
		FSFCNTRL	191	MR	209	215	007	
		FSFCNTRL	192	MR	216	218	003	
		FSFCNTRL	193	MR	219	221	003	
		FSFCNTRL	194	MR	222	223	002	
		FSFCNTRL	195	MR	224	225	002	
		FSFCNTRL	197	MR	226	226	001	
		FSFCNTRL	200	MR	227	227	001	
		FSFCNTRL	400	MR	228	228	001	
		FSFADMIN	192	MR	229	229	001	
		MAINT	324	MW	230	242	013	
		CSPUSER	191	MR	243	246	004	
		CSPUSER	193	MR	247	253	007	
		CSPUSER	502	MR	254	273	020	
		CSPUSER	503	MR	274	293	020	
					294	326	033	GAP
		$TEMP$	C03	R	327	426	100	
					427	884	458	GAP
VMSRES	3380	DIRMAINT	123	MW	000	884	885	
		MAINT	123	MW	000	884	885	
		SYSDUMP1	123	RR	000	884	885	
		$ALLOC$	A01	R	000	000	001	
		$SAVSYS$	A02	R	001	010	010	
		MAINT	191	MW	011	018	008	
		MAINT	194	MW	019	039	021	
		OPERATOR	191	MR	040	040	001	
					041	051	011	GAP
		MAINT	190	MW	052	096	045	
		$DIRECT$	A05	R	097	098	002	
		EREP	191	WR	099	099	001	
		MAINT	19E	MW	100	126	027	
		MAINT	319	MW	127	171	045	
		MAINT	196	MW	172	185	014	

Figure 66 (Part 2 of 3). Directory Map Sample for a 3380 Device

VOLSER	DEVTYPE	OWNERID	VADDR	MODE	START	END	LENGTH	FLAGS
VMSRES	3380				186	211	026	GAP
		OPERATNS	191	MR	212	212	001	
		OPERATNS	193	MR	213	220	008	
		MAINT	19D	MW	221	247	027	
		MAINT	296	MW	248	261	014	
					262	401	140	GAP
		$TEMP$	A04	R	402	441	040	
		$SYSCKP$	A06	R	442	442	001	
		$TEMP$	A09	R	443	482	040	
		$TDISK$	A10	R	483	582	100	
					583	831	249	GAP
		MAINT	201	MW	832	852	021	
		MAINT	293	MW	853	863	011	
		MAINT	294	MW	864	874	011	
		IVPM1	191	WR	875	875	001	
		IVPM2	191	WR	876	876	001	
		$SYSWRM$	A07	R	877	878	002	
		$SYSERR$	A06	R	879	880	002	
		$CPNUC$	A09	R	881	884	004	
VMSTGE	3380	MAINT	126	MW	000	884	885	
		SYSDUMP1	126	RR	000	884	885	
		$ALLOC$	D01	R	000	000	001	
		MAINT	393	WR	001	068	068	
		MAINT	396	WR	069	099	031	
					100	808	709	GAP
		MAINT	394	WR	809	884	076	

Figure 66 (Part 3 of 3). Directory Map Sample for a 3380 Device

3380 DMKRIO File Supplied With The Product Tape

```
RIO      TITLE 'DMKRIO - 3380    VM/SP'
DMKRIO   CSECT
         PRINT NOGEN
         COPY  OPTIONS
**********************************************************************
*
*       C L U S T E R    A N D    L I N E    C O N T R O L L E R S
*
**********************************************************************
         SPACE
CLUST056 CLUSTER  CUTYPE=3275,GPOLL=407F,LINE=056
         TERMINAL TERM=3275,SELECT=6040,MODEL=3
CLUST057 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=057
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3286,SELECT=60C6,MODEL=2
         TERMINAL TERM=3284,SELECT=60C7,MODEL=2
CLUST058 CLUSTER  CUTYPE=3271,GPOLL=407F,LINE=058
         TERMINAL TERM=3277,SELECT=6040,MODEL=2
         TERMINAL TERM=3277,SELECT=60C1,MODEL=2
         TERMINAL TERM=3277,SELECT=60C2,MODEL=2
         TERMINAL TERM=3277,SELECT=60C3,MODEL=2
         TERMINAL TERM=3277,SELECT=60C4,MODEL=2
         TERMINAL TERM=3277,SELECT=60C5,MODEL=2
         TERMINAL TERM=3277,SELECT=60C6,MODEL=2
         TERMINAL TERM=3277,SELECT=60C7,MODEL=2
       SPACE
       RDEVICE ADDRESS=040,DEVTYPE=3705,MODEL=H8,ADAPTER=TYPE4,
               CPTYPE=EP
       RDEVICE ADDRESS=(042,2),DEVTYPE=3705,ADAPTER=BSCA,
               BASEADD=040
       RDEVICE ADDRESS=056,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST056
       RDEVICE ADDRESS=057,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST057
       RDEVICE ADDRESS=058,DEVTYPE=3705,ADAPTER=BSCA,CLUSTER=CLUST058
       RDEVICE ADDRESS=05C,DEVTYPE=3705,ADAPTER=BSCA
       RDEVICE ADDRESS=(060,7),DEVTYPE=3705,ADAPTER=IBM1
       RDEVICE ADDRESS=067,DEVTYPE=3705,ADAPTER=TELE2
       RDEVICE ADDRESS=068,DEVTYPE=3705,ADAPTER=IBM1
       EJECT
```

Figure 67 (Part 1 of 4). Sample DMKRIO File for a 3380 Device

```
***********************************************************************
*
*                     C H A N N E L    Z E R O
*
* NOTE: 010 IS A SYSTEM CONSOLE ADDRESS FOR THE 4331
*            OR 4341 PROCESSOR.
*
*    NOTE: 014 MAY BE USED AS DISPLAY CONSOLE FOR 138, 148 AND 158.
*          (01A,2) AND (005,2) ARE THE 303X SYSTEM CONSOLES AND SRFS.
*           016 MAY BE USED AS DISPLAY CONSOLE ON 168 AND 165-II.
*
***********************************************************************
         SPACE
PRT      RDEVICE ADDRESS=002,DEVTYPE=3211,CLASS=(A,T),FEATURE=UNVCHSET
SPECDEV  RDEVICE ADDRESS=(005,2),DEVTYPE=7443
CONS     RDEVICE ADDRESS=009,DEVTYPE=3215
RDR      RDEVICE ADDRESS=00C,DEVTYPE=2540R
PUN      RDEVICE ADDRESS=00D,DEVTYPE=2540P,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=00E,DEVTYPE=1403,CLASS=(A,E),FEATURE=UNVCHSET
PRT      RDEVICE ADDRESS=00F,DEVTYPE=3203,MODEL=4,CLASS=A
CONS     RDEVICE ADDRESS=010,DEVTYPE=3278,MODEL=2A
PRT      RDEVICE ADDRESS=011,DEVTYPE=3262,CLASS=(A,T)
RDR      RDEVICE ADDRESS=012,DEVTYPE=3505
PUN      RDEVICE ADDRESS=013,DEVTYPE=3525,CLASS=(B,C,D,G)
CONS     RDEVICE ADDRESS=014,DEVTYPE=3148
CONSPRT  RDEVICE ADDRESS=015,DEVTYPE=3215
CONS     RDEVICE ADDRESS=016,DEVTYPE=3066
GRAF     RDEVICE ADDRESS=(018,2),DEVTYPE=3278,MODEL=2
CONS     RDEVICE ADDRESS=(01A,2),DEVTYPE=3036
PRT      RDEVICE ADDRESS=01E,DEVTYPE=3289E,MODEL=4,CLASS=(A,T)
CONS     RDEVICE ADDRESS=01F,DEVTYPE=3215
RDR      RDEVICE ADDRESS=02C,DEVTYPE=3505
PUN      RDEVICE ADDRESS=02D,DEVTYPE=3525,CLASS=(B,C,D,G)
PRT      RDEVICE ADDRESS=02E,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(030,16),DEVTYPE=3277
PRT      RDEVICE ADDRESS=0CE,DEVTYPE=3203,MODEL=5,FEATURE=UNVCHSET,
               CLASS=(A,C)
GRAF     RDEVICE ADDRESS=(0E0,16),DEVTYPE=3278,MODEL=2
         EJECT
***********************************************************************
*
*                     C H A N N E L    O N E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(100,8),DEVTYPE=FB-512
         RDEVICE ADDRESS=(120,32),DEVTYPE=3277
         RDEVICE ADDRESS=(140,16),DEVTYPE=3380
         RDEVICE ADDRESS=(150,16),DEVTYPE=3380
         RDEVICE ADDRESS=(160,16),DEVTYPE=3380
         RDEVICE ADDRESS=(170,16),DEVTYPE=3380
         RDEVICE ADDRESS=(180,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(190,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(1A0,16),DEVTYPE=3380
         EJECT
```

Figure 67 (Part 2 of 4). Sample DMKRIO File for a 3380 Device

```
***********************************************************************
*
*                     C H A N N E L   T W O
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(220,32),DEVTYPE=3277
         RDEVICE ADDRESS=(240,8),DEVTYPE=3340
         RDEVICE ADDRESS=(248,8),DEVTYPE=3350
         RDEVICE ADDRESS=(250,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(258,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(260,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(280,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(290,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(2C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=(2D0,16),DEVTYPE=3380
         EJECT
***********************************************************************
*
*                     C H A N N E L   T H R E E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(320,32),DEVTYPE=3277
         RDEVICE ADDRESS=(340,8),DEVTYPE=3340
         RDEVICE ADDRESS=(348,8),DEVTYPE=3350
         RDEVICE ADDRESS=(350,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(358,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(360,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(380,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(390,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(3C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=(3D0,16),DEVTYPE=3380
         EJECT
***********************************************************************
*
*                     C H A N N E L   F O U R
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(420,32),DEVTYPE=3277
         RDEVICE ADDRESS=(440,8),DEVTYPE=3340
         RDEVICE ADDRESS=(448,8),DEVTYPE=3350
         RDEVICE ADDRESS=(450,8),DEVTYPE=3330,MODEL=1
         RDEVICE ADDRESS=(458,8),DEVTYPE=3330,MODEL=11
         RDEVICE ADDRESS=(460,32),DEVTYPE=3278,MODEL=2
         RDEVICE ADDRESS=(480,8),DEVTYPE=3420,MODEL=8,FEATURE=DUALDENS
         RDEVICE ADDRESS=(490,8),DEVTYPE=2420,MODEL=7
         RDEVICE ADDRESS=(4C0,8),DEVTYPE=3340
         RDEVICE ADDRESS=4D0,DEVTYPE=3851
         RDEVICE ADDRESS=(4D8,4),DEVTYPE=3330,MODEL=11,FEATURE=VIRTUAL
         EJECT
***********************************************************************
*
*                     C H A N N E L   F I V E
*
***********************************************************************
         SPACE
         RDEVICE ADDRESS=(560,8),DEVTYPE=3420,MODEL=6,FEATURE=DUALDENS
         RDEVICE ADDRESS=(580,8),DEVTYPE=3420,MODEL=7,FEATURE=DUALDENS
         EJECT
```

Figure 67 (Part 3 of 4). Sample DMKRIO File for a 3380 Device

```
***********************************************************************
*
*                     C O N T R O L   U N I T S
*
***********************************************************************
         SPACE
RCTLU0   RCTLUNIT ADDRESS=000,CUTYPE=3811
         RCTLUNIT ADDRESS=008,CUTYPE=2821
         RCTLUNIT ADDRESS=010,CUTYPE=3148
         RCTLUNIT ADDRESS=018,CUTYPE=3505
         RCTLUNIT ADDRESS=028,CUTYPE=3505
         RCTLUNIT ADDRESS=030,CUTYPE=3272,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=040,CUTYPE=3705,FEATURE=64-DEVICE
         RCTLUNIT ADDRESS=0C8,CUTYPE=3203
         RCTLUNIT ADDRESS=0E0,CUTYPE=3274,FEATURE=32-DEVICE
RCTLU1   RCTLUNIT ADDRESS=100,CUTYPE=3880,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=120,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=140,CUTYPE=3880,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=160,CUTYPE=3880,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=180,CUTYPE=3803
         RCTLUNIT ADDRESS=190,CUTYPE=2803
         RCTLUNIT ADDRESS=1A0,CUTYPE=3880,FEATURE=16-DEVICE
RCTLU2   RCTLUNIT ADDRESS=220,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=240,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=260,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=280,CUTYPE=3803
         RCTLUNIT ADDRESS=290,CUTYPE=2803
         RCTLUNIT ADDRESS=2C0,CUTYPE=3830,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=2D0,CUTYPE=3880,FEATURE=16-DEVICE
RCTLU3   RCTLUNIT ADDRESS=320,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=340,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=360,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=380,CUTYPE=3803
         RCTLUNIT ADDRESS=390,CUTYPE=2803
         RCTLUNIT ADDRESS=3C0,CUTYPE=3830,FEATURE=16-DEVICE
         RCTLUNIT ADDRESS=3D0,CUTYPE=3880,FEATURE=16-DEVICE
RCTLU4   RCTLUNIT ADDRESS=420,CUTYPE=3272,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=440,CUTYPE=3830,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=460,CUTYPE=3274,FEATURE=32-DEVICE
         RCTLUNIT ADDRESS=480,CUTYPE=3803
         RCTLUNIT ADDRESS=490,CUTYPE=2803
         RCTLUNIT ADDRESS=4C0,CUTYPE=3830,FEATURE=32-DEVICE
RCTLU5   RCTLUNIT ADDRESS=560,CUTYPE=3803
         RCTLUNIT ADDRESS=580,CUTYPE=3803
         EJECT
***********************************************************************
*
*                         C H A N N E L S
*
***********************************************************************
         SPACE
         RCHANNEL ADDRESS=0,CHTYPE=MULTIPLEXOR
         RCHANNEL ADDRESS=1,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=2,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=3,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=4,CHTYPE=BLKMPXR
         RCHANNEL ADDRESS=5,CHTYPE=BLKMPXR
         EJECT
***********************************************************************
*
*                C O N S O L E
*
***********************************************************************
         SPACE
         RIOGEN  CONS=014,ALTCONS=(01F,009,01A,016,010)
         END
```

Figure 67 (Part 4 of 4). Sample DMKRIO File for a 3380 Device

3380 DMKSYS File Supplied With The Product Tape

The 3380 sample DMKSYS module supplied on the product tape is:

```
       PRINT   NOGEN
DMKSYS CSECT
       SYSOWN  VMSRES,                                                 X
               PROFPK,                                                 X
               VMPK01
       SYSRES  SYSVOL=VMSRES,                                          X
               SYSRES=123,                                             X
               SYSTYPE=3380,                                           X
               SYSCLR=YES,                                             X
               SYSNUC=881,                                             X
               SYSWRM=(877,2),                                         X
               SYSERR=(879,2),                                         X
               SYSCKP=(442,1)
       SYSMON  USERID=VMAP,                                            X
               AUTO=NO,                                                X
               BUFFS=3,                                                X
               TIME=(08:00,17:00),                                     X
               CLASS=M,                                                X
               ENABLE=(PERFORM,USER,DASTAP),                           X
               LIMIT=(50000,NOSTOP)
       SYSJRL
       SYSCOR  RMSIZE=4M,                                              X
               AP=NO
       SYSOPR  SYSOPER=OPERATOR,                                       X
               SYSDUMP=OPERATNS
       SYSACNT USERID=DISKACNT,                                        X
               OUTPUT=READER,                                          X
               CLASS=C,                                                X
               LIMIT=100
       SYSTIME ZONE=4,                                                 X
               LOC=WEST,                                               X
               ID=EDT
       SYSFORM
       SYSPCLAS
       SYSID
       SYSORD
       SYSMIH
       SYSLOCS
       END
```

Figure 68. Sample DMKSYS File for a 3380 Device

3380 DMKSNT File Supplied With The Product Tape

```
SNT      TITLE 'DMKSNT        VM/SP REL 3        3380 SAMPLE'
         SPACE
*.
* MODULE NAME -
*        DMKSNT
*
* FUNCTION -
*
*        THIS MODULE WILL BE ASSEMBLED BY THE INSTALLATION SYSTEM
*        PROGRAMMER. IT WILL DESCRIBE THE SYSTEM TO BE SAVED VIA
*        THE 'SAVESYS' COMMAND AND TO BE IPL'ED BY NAME. SHARED SEG-
*        MENTS MAY BE SPECIFIED. THESE SEGMENTS MUST CONSIST OF
*        ALL REENTRANT CODE.
*
*
*        INPUT TO THE NAMESYS MACRO IS SPECIFIED IN THE FOLLOWING
*        FORMAT:
*
*        LABEL NAMESYS    SYSSIZE=NNNK,SYSNAME=CCCCCC,VSYSRES=CCCCCC,
*                         VSYSADR=XXX,SYSVOL=CCCCCC,SYSCYL=NNN,
*                         SYSSTRT=(CC,P),SYSPGCT=NN,
*                         SYSPGNM=(NN,NN,NN-NN,......),
*                         SYSHRSG=(N,N,...)
*
*        WHERE:
*
*        SYSSIZE - THIS IS THE MINUMUM STORAGE SIZE NEEDED TO
*                  OPERATE THE SAVED SYSTEM.
*        SYSNAME - IS THE NAME GIVEN THE SYSTEM TO BE USED FOR
*                  IDENTIFICATION BY 'SAVESYS' AND 'IPL'.
*        VSYSRES - IS THE VOLUME SERIAL OF THE DASD CONTAINING THE
*                  SYSTEM TO BE SAVED
*        VSYSADR - IS THE VIRTUAL ADDRESS OF THE DASD CONTAINING
*                  THE SYSTEM.
*        SYSCYL  - THE CYLINDER ADDRESS OF THE 'MINI-DISK'
*                  FOR THE SYSTEM TO BE SAVED.
*        SYSVOL  - IS THE VOLUME SERIAL OF THE DASD DESIGNATED TO
*                  RECEIVE THE SAVED SYSTEM. THIS MUST BE A
*                 'CP-OWNED' VOLUME.
*        SYSSTRT - THIS DESIGNATES THE STARTING CYLINDER AND PAGE
*                  ADDRESS ON 'SYSVOL' THAT THIS NAMED SYSTEM IS TO
*                  BE SAVED. DURING THE SAVESYS AND IPL PROCESSING,
*                  THIS WILL BE USED TO MAKE UP THE 'CCPD' ADDRESS
*                  FOR THE DASD OPERATIONS. THESE NUMBERS ARE TO
*                  SPECIFIED IN DECIMAL.
*        SYSPGCT - IS THE TOTAL NUMBER OF PAGES TO BE SAVED.
*        SYSPGNM - THESE ARE THE NUMBERS OF THE PAGES TO BE SAVED.
*                  SPECIFICATION MAY BE DONE AS GROUPS OF PAGES OR
*                  AS SINGLE PAGES. FOR EXAMPLE - IF PAGES 0,4, AND
*                  10 THRU 13 ARE TO BE SAVED, USE THE FORMAT:
*                  SYSPGNM=(0,4,10-13).
*        SYSHRSG - THESE ARE THE SEGMENT NUMBERS DESIGNATED AS
*                  SHARED. THE PAGES IN THESE SEGMENTS WILL BE SET
*                  UP AT IPL TIME TO BE USED BE ANY USER
*                  IPL'ING BY THIS NAME.
```

Figure 69 (Part 1 of 8). Sample DMKSNT File for a 3380 Device

```
*
*        THE FOLLOWING IS A SAMPLE OF INPUT FOR THIS MODULE:
*
*        DMKSNTBL CSECT
*        FSTNAME NAMESYS   SYSSIZE=256K,SYSNAME=CMS,VSYSRES=CPDSK1,
*                          SYSVADR=190,SYSCYL=100,SYSVOL=CPDSK2,
*                          SYSSTRT=(400,1),SYSPGCT=10,
*                          SYSPGNM=(0-5,10-13),SYSHRSG=(1,2)
*
* ATTRIBUTES -
*
*        PAGEABLE
*
* ENTRY POINTS -
*
*        DMKSNTBL
*
* NOTES -
*
*        THERE IS NO EXECUTABLE CODE IN THIS MODULE.
*
*
*.
         EJECT
*
*********************************************************************
*
*        THE FOLLOWING ENTRIES ARE BASED ON THE INFORMATION PROVIDED
*        IN THE PLANNING GUIDE AND REFERENCE.
*
*********************************************************************
*
         SPACE
DMKSNTBL CSECT
         SPACE
*
*     HEX LOAD ADDRESS FOR SEGMENT  25 = 190000
*     THE SPACE FOR CMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*    ( THE ALLOCATIONS ARE BASED ON 150 PAGES/3380 CYLINDER )
*          CYL 1, PAGE 1 TO CYL 1, PAGE 138 (138 PAGES)
*         137 PAGES FOR CMS, 1 FOR CP INFORMATION.
*
*********************************************************************
CMS      NAMESYS    SYSNAME=CMS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(001,1),                                        X
               SYSPGNM=(0-4,14-33,400-511),                            X
               SYSPGCT=137,                                            X
               SYSHRSG=(25,26,27,28,29,30,31),                         X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=52,                                              X
               VSYSRES=VMSRES
         EJECT
```

Figure 69 (Part 2 of 8). Sample DMKSNT File for a 3380 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 240 = F00000
*     THE SPACE FOR CMSL IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 1, PAGE 139 TO CYL 2, PAGE 126 (138 PAGES)
*********************************************************************
CMSL      NAMESYS    SYSNAME=CMSL,                                     X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(01,139),                                       X
               SYSPGNM=(0-4,14-33,3840-3951),                          X
               SYSPGCT=137,                                            X
               SYSHRSG=(240,241,242,243,244,245,246),                  X
               SYSSIZE=256K,                                           X
               VSYSADR=190,                                            X
               SYSCYL=52,                                              X
               VSYSRES=VMSRES
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  49 = 310000
*     THE SPACE FOR CMSDOS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 2, PAGE 127 TO CYL 2, PAGE 143 (17 PAGES)
*********************************************************************
CMSDOS   NAMESYS SYSNAME=CMSDOS,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(02,127),                                       X
               SYSPGNM=(784-799),                                      X
               SYSPGCT=16,                                             X
               SYSHRSG=(49),                                           X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  50 = 320000
*     THE SPACE FOR CMSBAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 2, PAGE 144 TO CYL 3, PAGE 42 (49 PAGES)
********************************************************************
CMSBAM   NAMESYS SYSNAME=CMSBAM,                                       X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(02,144),                                       X
               SYSPGCT=48,                                             X
               SYSPGNM=(800-847),                                      X
               SYSHRSG=(50,51,52),                                     X
               SYSSIZE=192K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 69 (Part 3 of 8). Sample DMKSNT File for a 3380 Device

```
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  33 = 210000
*     THE SPACE FOR CMSVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 3, PAGE 43 TO CYL 4, PAGE 05 ( 113 PAGES )
*********************************************************************
CMSVSAM  NAMESYS SYSNAME=CMSVSAM,                                      X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(03,043),                                       X
               SYSPGNM=(528-639),                                      X
               SYSPGCT=112,                                            X
               SYSHRSG=(33,34,35,36,37,38),                            X
               SYSSIZE=448K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT  40 = 280000
*     THE SPACE FOR CMSAMS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 4, PAGE 06 TO CYL 4, PAGE 150 (145 PAGES)
*********************************************************************
CMSAMS  NAMESYS SYSNAME=CMSAMS,                                        X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(04,006),                                       X
               SYSPGNM=(640-783),                                      X
               SYSPGCT=144,                                            X
               SYSHRSG=(40,41,42,43,44,45),                            X
               SYSSIZE=576K,                                           X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*     HEX LOAD ADDRESS FOR SEGMENT 250 = FA0000
*     THE SPACE FOR INSTVSAM IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 5, PAGE 1 TO CYL 5, PAGE 17 (17 PAGES)
********************************************************************
INSTVSAM   NAMESYS SYSNAME=INSTVSAM,                                   X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(05,001),                                       X
               SYSPGNM=(4000-4015),                                    X
               SYSPGCT=16,                                             X
               SYSHRSG=(250),                                          X
               SYSSIZE=64K,                                            X
               SYSCYL=,                                                X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
*
         EJECT
```

Figure 69 (Part 4 of 8). Sample DMKSNT File for a 3380 Device

```
**********************************************************************
*     THE SPACE FOR VMEP01 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 5, PAGE 18 TO CYL 5, PAGE 34 (17 PAGES)
**********************************************************************
         SPACE
VMEP01 NAMENCP CPNAME=VMEP01,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(005,018),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
**********************************************************************
*     THE SPACE FOR VMEP02 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL65, PAGE 35 TO CYL 5, PAGE 51 (17 PAGES)
**********************************************************************
         SPACE
VMEP02 NAMENCP CPNAME=VMEP02,                                          X
               CPSIZE=48K,                                             X
               CPTYPE=EP,                                              X
               SYSSTRT=(005,035),                                      X
               SYSPGCT=16,                                             X
               SYSVOL=VMSRES
         EJECT
**********************************************************************
*  DCF (PROGRAM NO. 5748-XX9) - SCRIPT
*     HEX LOAD ADDRESS FOR SEGMENT  67 = 430000
*     THE SPACE FOR DSMSEG3 IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL  5, PAGE 52 TO CYL 5, PAGE 148 (97 PAGES)
**********************************************************************
         SPACE
DSMSEG3  NAMESYS SYSNAME=DSMSEG3,                                      X
               SYSSIZE=384K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(005,052),                                      X
               SYSPGCT=96,                                             X
               SYSPGNM=(1072-1167),                                    X
               SYSHRSG=(67,68,69,70,71,72),                            X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
```

Figure 69 (Part 5 of 8). Sample DMKSNT File for a 3380 Device

```
*********************************************************************
*  VM/VSE  (PROGRAM NO. 5748-MS1)- DIALOG MANAGER SUPPORT
*     HEX LOAD ADDRESS FOR SEGMENT  73 = 490000
*     THE SPACE FOR DTRSEG IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 5, PAGE 149 TO CYL 6, PAGE  15 (17 PAGES)
*********************************************************************
         SPACE
DTRSEG   NAMESYS SYSNAME=DTRSEG,                                       X
               SYSSIZE=64K,                                            X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(05,149),                                       X
               SYSPGCT=16,                                             X
               SYSPGNM=(1168-1183),                                    X
               SYSHRSG=(73),                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  ISPF/VM (PROGRAM NO. 5668-960)- DIALOG MANAGER FOR ISPF
*     HEX LOAD ADDRESS FOR SEGMENT 128 = 800000
*     THE SPACE FOR ISPDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 6, PAGE 16 TO CYL 6, PAGE  112 (97 PAGES)
*********************************************************************
         SPACE
ISPDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISPDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(06,016),                                       X
               SYSPGNM=(2048-2143),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(128,129,130,131,132,133)
         EJECT
*********************************************************************
*  ISPF/VM PDF (PROGRAM NO. 5668-172)- DIALOG MGR PRGM DEVELOPMENT
*     HEX LOAD ADDRESS FOR SEGMENT 134 = 860000
*     THE SPACE FOR ISRDCSS IS ALLOCATED ON VMSRES AS FOLLOWS:
*          CYL 6, PAGE 113 TO CYL 7, PAGE 59 (97 PAGES)
*********************************************************************
         SPACE
ISRDCSS NAMESYS SYSSIZE=384K,                                          X
               SYSNAME=ISRDCSS,                                        X
               SYSPGCT=96,                                             X
               SYSVOL=VMSRES,                                          X
               SYSSTRT=(06,113),                                       X
               SYSPGNM=(2144-2239),                                    X
               VSYSRES=,                                               X
               SYSCYL=,                                                X
               VSYSADR=IGNORE,                                         X
               SYSHRSG=(134,135,136,137,138,139)
         EJECT
```

Figure 69 (Part 6 of 8). Sample DMKSNT File for a 3380 Device

```
*********************************************************************
*  IIPS       (PROGRAM NO. 5668-012)- COMPUTER BASED TRAINING PP
*     HEX LOAD ADDRESS FOR SEGMENT  54 = 360000
*     THE SPACE FOR IIS IS ALLOCATED ON VMSRES, AS FOLLOWS:
*          CYL 7, PAGE 60 TO CYL 7, PAGE 124 (65 PAGES)
*********************************************************************
         SPACE
IISDCSS  NAMESYS SYSNAME=IISDCSS,                                      X
               SYSSIZE=256K,                                           X
               SYSVOL=VMSRES,                                          X
               SYSCYL=,                                                X
               SYSSTRT=(07,60),                                        X
               SYSPGCT=64,                                             X
               SYSPGNM=(864-927),                                      X
               SYSHRSG=(54,55,56,57),                                  X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  VS APL 4.0 (PROGRAM NO. 5748-AP1)
*     HEX LOAD ADDRESS FOR SEGMENT  58 = 3A0000
*     THE SPACE FOR APLSYS1 IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 1, PAGE 97 (97 PAGES)
*********************************************************************
         SPACE
APLSYS1  NAMESYS SYSNAME=APLSYS1,                                      X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=320K,                                           X
               SYSHRSG=(58,59,60,61,62,63),                            X
               SYSPGNM=(928-1023),                                     X
               SYSPGCT=96,                                             X
               SYSSTRT=(001,01),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT  96 = 600000
*     THE SPACE FOR BASSEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 3, PAGE 89 TO CYL 4, PAGE 35 (97 PAGES)
*********************************************************************
BASSEG   NAMESYS SYSSIZE=384K,SYSNAME=BASSEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(003,89),SYSPGCT=96,              X
               SYSPGNM=(1536-1631),                                    X
               SYSHRSG=(96,97,98,99,100,101)
         EJECT
*********************************************************************
*  IBM BASIC (PROGRAM NO. 5668-996)
*     HEX LOAD ADDRESS FOR SEGMENT 102 = 660000
*     THE SPACE FOR BLISEG  IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 4, PAGE  36 TO CYL 4, PAGE  84 (49 PAGES)
*********************************************************************
BLISEG   NAMESYS SYSSIZE=192K,SYSNAME=BLISEG,VSYSRES=,VSYSADR=IGNORE,  X
               SYSVOL=VMPK01,SYSSTRT=(004,036),                        X
               SYSPGCT=48,SYSPGNM=(1632-1679),                         X
               SYSHRSG=(102,103,104)
         EJECT
```

Figure 69 (Part 7 of 8). Sample DMKSNT File for a 3380 Device

```
**********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 4, PAGE 85 TO CYL 6, PAGE  25 (241 PAGES)
**********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(04,85),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
**********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 6, PAGE  26 TO CYL 9, PAGE  24 (449 PAGES)
**********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(06,026),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
**********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE  43 (193 PAGES)
**********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 69 (Part 8 of 8). Sample DMKSNT File for a 3380 Device

VMSRES Pack Layout for 3380 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-18	8	191 minidisk for MAINT
19-39	21	194 minidisk for MAINT (CP)
40	1	191 minidisk for OPERATOR
41-51	11	xxx minidisk for user option
52-96	45	190 minidisk for MAINT (CMS system disk)*
97-98	2	VM/SP directory
99	1	191 minidisk for EREP
100-126	27	19E minidisk for MAINT
127-171	45	319 minidisk for MAINT
172-211	40	xxx minidisk for user option
212	1	191 minidisk for OPERATNS
213-220	8	193 minidisk for OPERATNS
221-247	27	19D minidisk for MAINT (HELP)
248-401	154	xxx minidisk for user option
402-441	40	Spool space
442	1	Spool file checkpoint
443-482	40	Spool space
483-582	100	T-DISK space
583-831	249	xxx minidisk for user option
832-852	21	201 minidisk for MAINT
853-863	11	293 minidisk for MAINT (CMS service)
864-874	11	294 minidisk for MAINT (CP service)
875	1	191 minidisk for IVPM1
876	1	191 minidisk for IVPM2
877-878	2	Warm start data
879-880	2	I/O Error Recording area
881-884	4	CP nucleus

Figure 70. VMSRES Layout for a 3380 Starter System

* Specify cylinder 43 to load the CMS nucleus which occupies the first 1 cylinder of the recompiled area.
Specify cylinder 44 to load the CMSL nucleus which occupies the last 1 cylinder of the recompiled area.

Glossary

abend dump. To write the contents of main storage, or of part of main storage, to an external medium for the purpose of debugging an error condition that resulted in the termination of a task prior to its normal completion.

access mode. A mode that determines whether the file can be used as read-only or read/write.

alphameric. Pertaining to a character set that contains letters, digits, and usually other characters, such as punctuation marks.

alternate console. A console assigned as a backup unit to the system console.

APAR. Authorized program analysis report. A report of a problem caused by a suspected defect in a current unaltered release of a program.

attention interrupt. An I/O interruption caused by a terminal user pressing an attention key, or its equivalent.

attention key. A function key on terminals that, when pressed, causes an I/O interruption in the processing unit.

auxiliary control file. A file that contains a list of filetypes of update files to be applied to a particular source file. See also control file.

auxiliary directory. An extension of the CMS file directory, which contains the names and locations of certain CMS modules that are not included in the CMS file directory.

block extent. A continuous space on a direct access storage volume that does not have to be stored contiguously with a block (4096) of pages.

bootstrap program. A technique or device designed to bring itself into a desired state by means of its own action, for example, a machine routine whose first few instructions are sufficient to bring the rest of itself into the computer from an input device.

bpi. Bits per inch.

channel. A device that connects a processor and main storage with I/O control units.

CKD device. Count-key-data device. A disk storage device that stores data in the format: count field, usually followed by a key field, followed by the actual data of a record.

CKPT. See checkpoint start.

checkpoint start. A system restart that attempts to recover information about closed spool files that were previously stored on the checkpoint cylinders. The spool file chains are reconstructed, but the original sequence of spool files is lost. Unlike warm start, CP accounting and system message information is also lost. Contrast with warm start, cold start, force start.

class authority. Privilege assigned to a virtual machine user in the user's directory entry; each class specified allows access to a logical subset of all the CP commands.

CMS. Conversational monitor system. A virtual machine operating system that provides general interactive time sharing, problem solving, and program development capabilities, and operates only under the control of the VM/SP control program.

CMS file system. A means by which to create files in the CMS system. CMS disk files are created by using an identifier consisting of three fields: filename, filetype, and filemode. These files are unique to the CMS system and cannot be read or written using other operating systems.

CMS FORMAT command. The command issued to format disks which are to contain CMS files. As an option, this command allows for a choice of physical disk block sizes.

CMS loader work file. The CMS file loaded into storage by the LOAD command.

CMS nucleus. The portion of CMS that is resident in main storage.

CMS service minidisk. The minidisk located at virtual address 190. It contains the system commands for which you have read only access.

CNTRL file. See control file.

COLD start. A system restart that ignores previous data areas and accounting information in main storage, and the contents of paging and spool files on CP-owned disks.

communications scanner. A communication controller hardware unit that provides the connection between lines and the central control unit. The communications scanner monitors telecommunication lines and data links for service requests.

console. A device used for communication between the operator or maintenance engineer and the computer.

control block. A storage area used by a computer program to hold control information.

control file. The file that contains records that identify the updates to be applied and the macro libraries, if any, that are needed to assemble a source program. See also auxiliary control file.

control statement. A statement that controls or affects the execution of a program in a data processing system.

control unit. A device that controls input/output operations at one or more devices.

CP. VM/SP control program. The component of VM/SP that manages the resources of a single computer so that multiple computing systems appear to exist.

cylinder. In a disk pack, the set of all tracks with the same nominal distance from the axis about which the disk pack rotates.

DASD. Direct access storage device. A device in which the access time is effectively independent of the location of the data.

DASD dump restore (DDR) program. A service program used to copy all or part of a minidisk onto tape, or to load the contents of a tape onto a minidisk.

DDR. See DASD dump restore (DDR) program.

directory. VM/SP directory. A CP disk file that defines each virtual machine's normal configuration.

display device. An input/output device that gives a visual representation of data.

display terminal. A terminal with a component capable of displaying information on a viewing surface such as a cathode ray tube or gas panel.

dump. See abend dump.

EC level. Engineering change level.

ECMODE. A mode in which all the features of a System/370 computing system, including dynamic address translation, are operational.

emulation. The use of programming techniques and special machine features to permit a computing system to execute programs written for another system.

emulation program (EP). A control program that allows a 3704 or 3705 Communications Controller to emulate the functions of an IBM 2701 Data Adapter Unit, an IBM 2702 Transmission Control, or an IBM 2703 Transmission Control.

entry point. The address or label of the first instruction executed upon entering a computer program, a routine, or a subroutine.

EREP file. A collection of error records produced by the malfunctioning of hardware components on a computer and stored for processing by the environmental recording, editing, and printing (EREP) program.

filename. A name (up to eight alphameric characters) assigned to a CMS file.

forms control buffer (FCB). In the 3800 Printing Subsystem, a buffer for controlling the vertical format of printed output.

free storage. Storage that is not allocated.

full screen editor. An editor used at a display terminal where an entire screen of data is displayed at once and where the user can access the data through commands or by using a cursor.

IML. Initial microprogram load.

instruction counter. A counter that indicates the location of the next computer instruction to be interpreted.

IPL. Initial program load.

IVP. Installation verification program. A program used to verify the correct installation of VM/SP.

load map. A map containing the storage addresses of control sections and entry points of a program loaded into storage.

loader. A routine, commonly a computer program, that reads data into main storage.

local update. A customer's own modifications or enhancements to VM/SP modules or source code during the generation of of the CP or CMS system.

macro. In assembler language programming, an assembler language statement that causes the assembler to process a predefined set of statements called a macro definition. The statements normally produced from the macro definition replace the macro instruction in the program.

macro library. A library of macro definitions.

master file directory. A directory on each CMS disk that contains the name, format, size, and location of all the CMS files on the disk. When a disk is accessed via the ACCESS command, the directory is read into main storage and identified with one of the twenty six disk mode letters (A-Z).

megabyte. 1,048,576 bytes

merged product tape. The tape that contains all current VM code. It is "merged" because it includes both VM/SP and VM/370 release 6 modules and macros. Most of this book is devoted to installation using the VM/SP Merged Product Tape.

minidisk. A logical subdivision (or all) of a physical disk pack that has its own virtual device address, consecutive virtual cylinders (starting with virtual cylinder zero), and a VTOC or disk label identifier. Each user virtual disk is preallocated and is defined via a VM/SP directory entry as belonging to a user.

module. A program unit that is discrete and identifiable with respect to compiling, combining with other units, and loading; for example, the input to, or output from, an assembler, compiler, linkage editor, or executive routine.

MP support. See multiprocessor.

multiprocessor. A computer employing two or more processing units under integrated control.

node. In a network, a point where one or more functional units interconnect transmission lines.

nucleus. That part of a control program that is resident in main storage.

object module. A module that is the output of an assembler or a compiler and is input to a linkage editor.

operand. Information entered with a command name to define the data on which a command processor operates and to control the execution of the command processor.

operators console. A functional unit containing devices that are used for communication between a computer operator and an automatic data processing system.

Optional Feature Products Tape. The tape that contains the Optional Feature Program Products that you will install with your system using INSTFPP.

overlay. The technique of repeatedly using the same areas of internal storage during different stages of a program.

pack. A set of flat, circular recording surfaces used in a disk storage device. A disk pack.

page. A fixed-length block that has a virtual address and that can be transferred between real storage and auxiliary storage.

page number. The part of a virtual storage address needed to refer to a page.

paging. The process of transferring pages between real storage and external page storage.

parameter. A variable that is given a constant value for a specified application and that may denote the application.

prompt. A displayed message that assists a terminal operator by describing required input or by giving operational information.

PSW (program status word). An area in storage used to indicate the order in which instructions are executed, and to hold and indicate the status of the computer system.

PTF (program temporary fix). A temporary solution or bypass of a problem diagnosed by IBM field engineering as the result of a defect in a current unaltered release of a program.

spooling devices. Input/output devices (card readers, punches, printers, DASD) used for reading input and writing output.

starter system tape. The tape you start with when installing "from scratch." Starter system tapes are DASD-type specific; for instance, you can't use "3350 Starter System" on any other DASD-type than 3350. Once restored, the Starter System is a very basic VM system which you use to build your own production system.

stand-alone. Pertaining to an operation that is independent of another device, program, or system.

SVC. Supervisor call instruction.

system name table. In the VM/SP control program the table that contains the name and location of saved systems, including discontiguous shared and non-shared segments.

virtual machine. A functional simulation of a computer and its associated devices.

VM/SP product tape. The tape that contains only those modules and macros added or altered since VM/370 release 6.

VMPK01. The label given to the disk pack that is required for installation using the Starter System. PREP EXEC and GENERATE EXEC use the TEMP and T-DISK space allocated on this pack for work space.

VMSEXT. The label given to the disk pack that is used for additional work space, or to hold programs if a 3310, 3330, or 3340 DASD is used as the system residence device. This pack is used but not required for the installation procedure documented in "Chapter 2. Installation When Using the Starter System Tape" on page 15.

VMSRES. The label given to the disk pack that holds the contents of the Starter System Tape and the Product Tape until the starter system portion of the installation procedure completes. The starter system, then, becomes the operational system.

VMSTGE. The label given to the disk pack used to hold CP source code for all DASD and CMS source code for 3330, 3350, 3370, 3375, and 3380 DASD. If your installation does not require source code to be maintained on DASD, this pack is not required.

VMSTG2. The label given to the disk pack used to hold CMS source code for 3310 and 3340 DASD only. If your installation does not require source code to be maintained on DASD, this pack is not required.

warm start. The result of an IPL that does not erase previous system data. The automatic reinitializing of the VM/SP control program that occurs if the control program cannot continue processing. Closed spool files and the VM/SP accounting information are not lost.

Index

SC24-5237-1

VM/SP Installation Guide (File No. S370/4300-34) Printed in U.S.A. SC24-5237-1

VM/SP
Installation Guide
SC24-5237-1

READER'S COMMENT FORM

This manual is part of a library that serves as a reference source for systems analysts, programmers, and operators of IBM systems. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

Your comments will be sent to the author's department for whatever review and action, if any, are deemed appropriate. Comments may be written in your own language; English is not required.

Note: *Copies of IBM publications are not stocked at the location to which this form is addressed. Please direct any requests for copies of publications, or for assistance in using your IBM system, to your IBM representative or to the IBM branch office serving your locality.*

	Yes	*No*
• Does the publication meet your needs?	□	□
• Did you find the material:		
Easy to read and understand?	□	□
Organized for convenient use?	□	□
Complete?	□	□
Well illustrated?	□	□
Written for your technical level?	□	□

• What is your occupation? ______________________________

• How do you use this publication:

As an introduction to the subject?	□	As an instructor in class?	□
For advanced knowledge of the subject?	□	As a student in class?	□
To learn about operating procedures?	□	As a reference manual?	□

Your comments:

If you would like a reply, please supply your name and address on the reverse side of this form.

Thank you for your cooperation. No postage stamp necessary if mailed in the U.S.A. (Elsewhere, an IBM office or representative will be happy to forward your comments or you may mail directly to the address in the Edition Notice on the back of the title page.)

Note: Staples can cause problems with automated mail sorting equipment.
Please use pressure sensitive or other gummed tape to seal this form.

Reader's Comment Form

Cut or Fold Along Line

Fold and Tape Please Do Not Staple Fold and Tape

NO POSTAGE
NECESSARY
IF MAILED
IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS PERMIT NO. 40 ARMONK, N.Y.

POSTAGE WILL BE PAID BY ADDRESSEE:

International Business Machines Corporation
Department G60
P. O. Box 6
Endicott, New York 13760

Fold Fold

If you would like a reply, *please print:*

Your Name ______________________________
Company Name ______________________ *Department* __________
Street Address ______________________________
City ______________________________
State ______________________ *Zip Code* __________
IBM Branch Office serving you ______________________________

IBM®

IBM Technical Newsletter

This Newsletter No. SN24-5750
Date August 31, 1984

Base Publication No. SC24-5237-1
File No. S370/4300-34

Previous Newsletters None

Virtual Machine/System Product Installation Guide

This Technical Newsletter, applies to Release 3.1, of the *IBM Virtual Machine/System Product Installation Guide* (Program Number 5664-167). It provides replacement and additional pages for your publication. These pages remain in effect until specifically altered. Pages to be added or replaced are:

Title Page, Notices
iii-viii
1, 2
15-30
35, 36
43, 44
51, 52
55, 56
56.1, 56.2
57-68
71, 72
72.1, 72.2
73-76
83-88
88.1, 88.2
95-98
103-110
119-128
128.1, 128.2
129, 130
130.1, 130.2
131-134
134.1, 134.2
141-144
165-168
183, 184
193-196
213-216
231-234
234.1, 234.2
245, 246
257-260
285, 286
289, 290
301, 302
331, 332
361, 362
362.1, 362.2
393, 394
423, 424
424.1, 424.2
453, 454
454.1, 454.2
485, 486
515, 516
516.1, 516.2

A change to the text is indicated by a vertical line to the left of the change.

Summary of Changes

This Technical Newsletter incorporates document changes due to unresolved problems with the lack of references to High Performance Option, the disparity between GENERATE EXEC code and documentation, and notices affecting system integrity.

For a detailed list of changes, see page iii.

Note: *Please insert this page in your publication to provide a record of changes.*

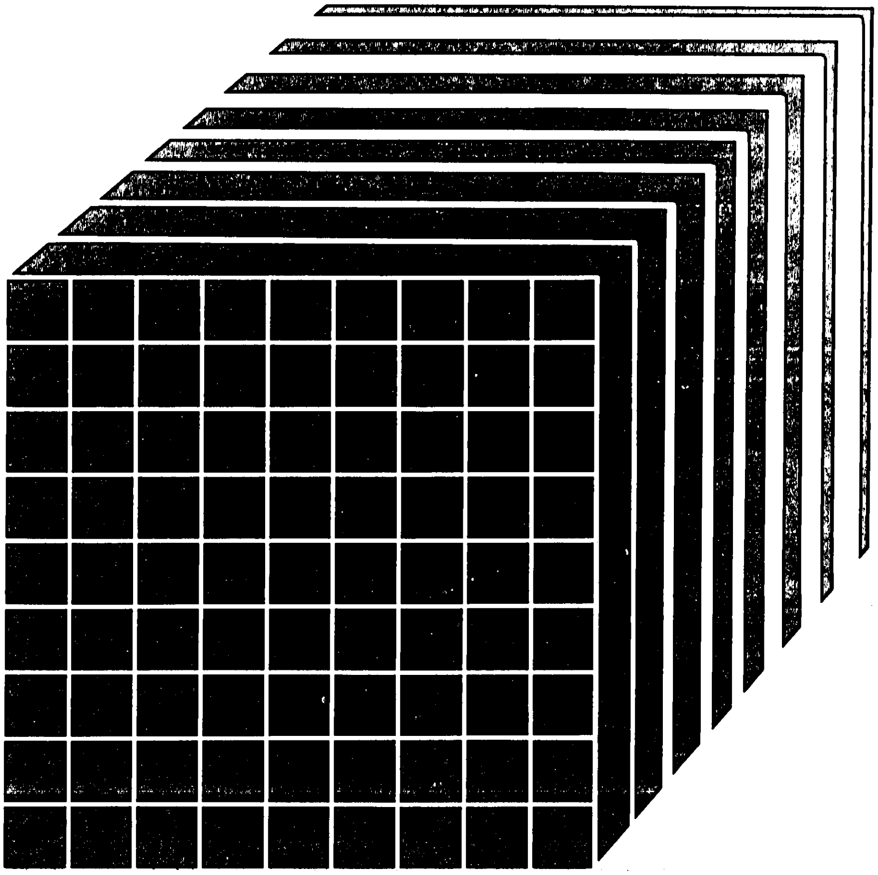
IBM Corporation, Programming Publications, Dept. G60, P.O. Box 6, Endicott, N.Y. 13760

Printed in U. S. A.

Virtual Machine/ System Product

Installation Guide

Release 3.1



Second Edition (February 1984)

This edition, SC24-5237-1, and Technical Newsletter SN24-5750, is a major revision of SC24-5237-0, and applies to Release 3, modification level 1 (Release 3.1) Virtual Machine/System Product (VM/SP), program number 5664-167, and to all subsequent releases and modifications until otherwise indicated in new editions or Technical Newsletters.

Summary of Changes

For a list of changes, see page iii.

Technical changes and additions to text and illustrations are indicated by a vertical line to the left of the change.

The Appendixes J through P contain listings of the sample files used with this release. The changes in the sample file listings are not identified by the vertical line.

Changes are made periodically to the information contained herein; before using this publication in connection with the operation of IBM systems, consult the latest *IBM System/370 and 4300 Processors Bibliography*, GC20-0001, for the editions that are applicable and current.

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 program product in this publication is not intended to state or imply that only IBM's program product may be used. Any functionally equivalent program may be used instead.

Publications are not stocked at the address given below. Requests for IBM publications should be made to your IBM representative or to the IBM branch office serving your locality.

A form for reader's comments is provided at the back of this publication. If the form has been removed, comments may be addressed to IBM Corporation, Programming Publications, Department G60, P.O. Box 6, Endicott, NY, U.S.A. 13760. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligation to you.

Summary of Changes

Summary of Changes
for SN24-5750
for VM/SP Release 3.1

Added:

- Documentation
  - HPO installation references to *VM/SP HPO Installation Guide*, SC38-0107
  - I/O references to *Input/Output Configuration Program User's Guide and Reference*, GC28-1027
- VMFPLC2 information
  - Option D to SCAN operand
  - Option F to LOAD operand
  - Messages and codes
- Where applicable, system integrity statements regarding use of system provided passwords.

Changed:

- Format for Chapter 6 to improve ease of use and readability
- Selective responses from GENERATE EXEC reflecting technical changes
- Minor technical and editorial changes.

Deleted:

- Sample allocations tables on pages 22 and 23 and placed in appropriate Appendix J through P

Summary of Changes
for SC24-5237-1
for VM/SP Release 3.1

Optional Feature Program Product Installation

New: Chapter 6 and Appendix I contain information for installing the optional feature program products.

Three new EXECs are added for this release. They are: PPPREP, INSTFPP, and DISKMAP.

Changed: The GENERATE EXEC is changed for this release.

Installation Guide Reorganization

Changed: The sample files are in Appendixes J through P. Chapter numbers after chapter 5 have been changed because of the addition of Chapter 6 and the relocation of the sample files.

Miscellaneous

Minor technical and editorial changes.

Summary of Changes
for SC24-5237-0
for VM/SP Release 3

- The CMSSEG Saved Segment has been removed. Code from CMSSEG has been incorporated into the CMS Nucleus.
- The CMSXGEN Procedure which formerly generated CMSSEG has been deleted.
- EREP files have been removed from the Product Tape.
- Space on the 190 minidisk has been reserved for the CMSL Nucleus.

The Sources for the SC24-5237-0 edition of this Book

This book includes material that originally appeared in the *VM/SP Planning and System Generation Guide.* (The *VM/SP: Planning and System Generation Guide* has been restructured to include planning information only. It has been retitled *VM/SP Planning Guide and Reference.*)

Most of this book came from Parts 3 through 5 and Appendixes C, F, and G of the *VM/SP Planning and System Generation Guide.*

- Chapters 1 through 5 of this book were taken from "Part 3. Generating VM/SP (CP and CMS)" of the *VM/SP Planning and System Generation Guide* and "Chapter 4. Installation Procedures Using the IPO1 Tape" of the *Universal Program Directory for VM/SP System IPO/E Release 2.1.*
- Chapters 6 through 12 of this book were created from sample files previously listed in "Chapter 23. Creating Your VM/SP Directory," "Chapter 24. Preparing System Name Table File (DMKSNT)," and "Chapter 27. Sample System Generation Using a Starter System" of the *VM/SP Planning and System Generation Guide* and from Appendixes A through F of the *VM/System Product System Installation Productivity Option/Extended: Planning Guide.*
- Chapters 13 through 16 of this book were taken from "Part 5. Updating VM/SP" of the *VM/SP Planning and System Generation Guide.*
- Appendix A of this book was taken from "Appendix C. CP/CMS Nucleus/Module Regeneration Requirements" of the *VM/SP Planning and System Generation Guide.*
- Appendix B of this book was taken from "Appendix F. A Sample EXEC Procedure for Copying VSE Macros into a CMS MACLIB" of the *VM/SP Planning and System Generation Guide.*
- Appendix C of this book was taken from "Appendix G. Generating VM/SP Without a VM/SP Starter System or the Merged Product Tapes" of the *VM/SP Planning and System Generation Guide.*
- Appendix D of this book was taken from "Chapter 33. Generating and Loading the 3704/3705 Control Program" of the *VM/SP Planning and System Generation Guide.*
- Appendix E of this book was taken from "Chapter 27. Sample System Generation Using a Starter System" of the *VM/SP Planning and System Generation Guide.*
- Appendix F of this book has been added as a result of an authorized program analysis report (APAR).
- Minor technical and editorial changes are also included.

Preface

The Intent of This Book

The *VM/SP Installation Guide* discusses how to install and service IBM Virtual Machine System Product (VM/SP). The necessary material, procedures, and examples are described in this book.

A general understanding of System/370 data processing and teleprocessing techniques is assumed. Also, a review of the contents of *VM/SP Introduction*, *VM/SP Planning Guide and Reference*, *VM/SP System Product Editor User's Guide*, and *Input/Output Configuration Program User's Guide and Reference* is required before this book is used.

This book has two parts, plus appendixes.

"Part 1. VM/SP Installation" describes the step-by-step procedure for installing CP and CMS. Two separate procedures are presented. The first is intended for those of you who wish to install VM/SP using both the Starter System and Product Tapes. The other is meant to be used by current VM/SP users who wish to install only the Product Tape using their existing VM/SP system.

Part 1 also includes a procedure for verifying CP and CMS, a method to load and save saved segments, and the procedure for installing the optional feature program products.

"Part 2. VM/SP Service" describes the procedures, programs, and EXECs used to update VM/SP source code and macro libraries.

Also included are appendixes about:

- CP/CMS nucleus/module regeneration requirements
- A sample EXEC procedure to copy VSE macros into a CMS MACLIB
- Installing VM/SP without both distribution tapes
- Generating the 3704/3705 control program
- Alternate CMS nucleus placement
- Creating additional CMS segment containing Y-stats
- Special options for CP
- Recording references for installation
- A list of the program products contained on the Feature tape(s) and considerations for installing the program products
- Sample file listings of configurations by device type.

In this book, the following terms have extended meanings:

- The term "Merged Product Tape" or just "Product Tape" refers to the tape which contains VM/SP and VM/370 Release 6 merged. It contains all of the CP, CMS and HELP files necessary to generate a complete VM/SP system including those required from VM/370 Release 6.

- The term "VM/SP Only Product Tape" refers to the tape which contains VM/SP modules only. This tape contains only the CP and CMS modules and macros that have been changed or added since VM/370 Release 6. (Often it is referred to as an unmerged product tape.)

- The term "3330 series" refers to the IBM 3330 Disk Storage Models 1 and 11; and the IBM 3333 Disk Storage and Control, Models 1 and 11.

- The term "3340 series" refers to the IBM 3340 Disk Storage, Models A2, B1 and B2, and the 3344 Direct Access Storage Model B2.

- The term "3350 series" refers to the IBM 3350 Direct Access Storage Models A2 and B2 in native mode.

- The term "3375" refers to the IBM 3375 Direct Access Storage Device.

- The term "3380" refers to the IBM 3380 Direct Access Storage Device.

- The term "FB-512" refers to the IBM 3310 and 3370 Direct Access Storage Devices.

- The term "3705" refers to the IBM 3705-I and 3705-II Communications Controllers, unless otherwise specified.

- The term "3270" refers to all VM/SP supported virtual machine display consoles unless otherwise noted. A specific device type is used only when it is necessary to show that device types differ.

- Information about display terminal usage also applies to the IBM 3138, 3148 and 3158 Display Consoles in display mode, unless otherwise noted.

- The term "display device" refers to any VM/SP supported system console terminal that displays data on a screen.

- Unless otherwise noted, the term "ATTN key" implies "(or equivalent)." It refers to the "signal interrupt control key" as listed for each terminal or console type in the *VM/SP Terminal Reference.* (For example, on the 3278 terminal it is the PA1 key.)

- Unless otherwise noted, the term VSE refers to the DOS/VSE system control program and the VSE/Advanced Functions program products combined.

  In certain cases, the term DOS is still used as a generic term. For example, disk packs initialized for use with VSE or any prior DOS or DOS/VS system may be referred to as DOS disks.

  The DOS-like simulation environment provided under the CMS portion of VM/SP continues to be referred to as CMS/DOS.

- CMS/DOS is part of the CMS system and is not a separate system. The term "CMS/DOS" states concisely that the VSE simulation mode of CMS is currently active. In other words, the CMS command **set dos on** already has been invoked.

- The phrase "the CMS file system" refers to disk files that are in CMS's 1K, 2K, or 4K fixed physical block format. CMS's VSAM data sets are not included.

- The phrase "System Product Editor" implies the use of the XEDIT command, XEDIT subcommands and macros, and XEDIT prefix subcommands.

The Contents of This Book

This book contains descriptions of the magnetic tapes and disk packs required to generate VM/SP. Step-by-step procedures for using them are also included. The descriptions and procedures show you how to:

- Format the DASD needed to install a system
- Install a new VM/SP system using the Starter System Tape
- Install VM/SP when not using the Starter System Tape
- Tailor the new system to specific needs
- Check that the newly installed CP and CMS are working properly
- Display sample directories, product tape files, and pack layouts
- Load and save discontiguous saved segments
- Update VM/SP thorough recommended service procedures
- Use EXEC procedures and commands.

Whether you are a new or existing user of VM/SP, you will find this book an important installation aid. You may use this guide to install the system on several types of direct access storage devices (DASD). Included are instructions for using the following devices: 3310, 3330, 3340, 3350, 3370, 3375, and 3380.

Corequisite Publications

- For information about the VM/SP system, the following publications are corequisite:

  Virtual Machine/System Product:

  Introduction, GC19-6200
  Planning Guide and Reference, SC19-6201
  Operator's Guide, SC19-6202
  System Programmer's Guide, SC19-6203
  System Messages and Codes, SC19-6204
  Terminal Reference, GC19-6206
  CMS Command and Macro Reference, GC19-6209
  CMS User's Guide, GC19-6210
  CP Command Reference for General Users, GC19-6211
  System Product Editor Command and Macro Reference, SC24-5221
  System Product Interpreter Reference, SC24-5239

  References in the text to titles of VM/SP books are given in abbreviated form.

- For information regarding VM/SP HPO installation procedures, refer to *VM/SP HPO Installation Guide*, SC38-0107.
- For information regarding 3081 devices, the following publication is corequisite: *Input/Output Configuration Program User's Guide and Reference*, GC28-1027.

- For information regarding 3270 display systems, the following publication is corequisite:

  3270 Information Display System Library User's Guide, GA23-0058

- For information concerning SNA supported devices, refer to the following publication:

  VM/VCNA Installation, Operations, and Terminal Use, SC27-0502

- If the IBM 3850 Mass Storage System is attached to the VM/SP processor, the following publications are corequisite:

  IBM 3850 Mass Storage System (MSS) Installation Planning and Table Create, GC35-0028

- For information regarding 3704/3705 Communications Controllers, the following publications are corequisite:

  IBM 3704 and 3705 Control Program Generation and Utilities Guide and Reference Manual (OS/VS TCAM Levels 5 and 6 in VS1; VS2 rel 1.6, 1.7, 2, SCP 5744-BA1, GC30-3007

  IBM 3704 and 3705 Control Program Generation and Utilities Guide and Reference Manual (TCAM 10 SVS - 5742-017) SCP 5742, 5744-AN1/BA2, 5747-AG | -AG2, GC30-3008

  Guide to Using the IBM 3704 Communications Controller Control Panel, GA27-3086

  Guide to Using the IBM 3705 Control Panel, GA27-3087

- For information regarding DASD support, the following publication is corequisite:

  Device Support Facility User's Guide and Reference, GC35-0033

Part 1. VM/SP Installation

Throughout this book you will see references to three types of tapes. The tapes and the information they contain are:

Starter System Tape

The tape you start with when installing "from scratch." Starter system tapes are DASD-type specific; for instance, you can't use "3350 Starter System" on any other DASD-type than 3350. Once restored, the Starter System is a very basic VM system which you use to build your own production system.

Note: The Starter System is NOT a production system; its only use is installing the VM/SP Merged Product Tape.

VM/SP Product Tape

VM/SP is shipped in two different ways, depending on the feature codes you specify in your order:

"Merged Product Tape" contains all current VM code. It is "merged" because it includes both VM/SP and VM/370 release 6 modules and macros. Most of this book is devoted to installation using the VM/SP Merged Product Tape.

"VM/SP Product Tape" contains only those modules and macros added or altered since VM/370 release 6. Installation using this tape is detailed in Appendix C, "VM/SP Without the Starter System and the Merged Product Tapes" on page 251.

Optional Feature Products Tape

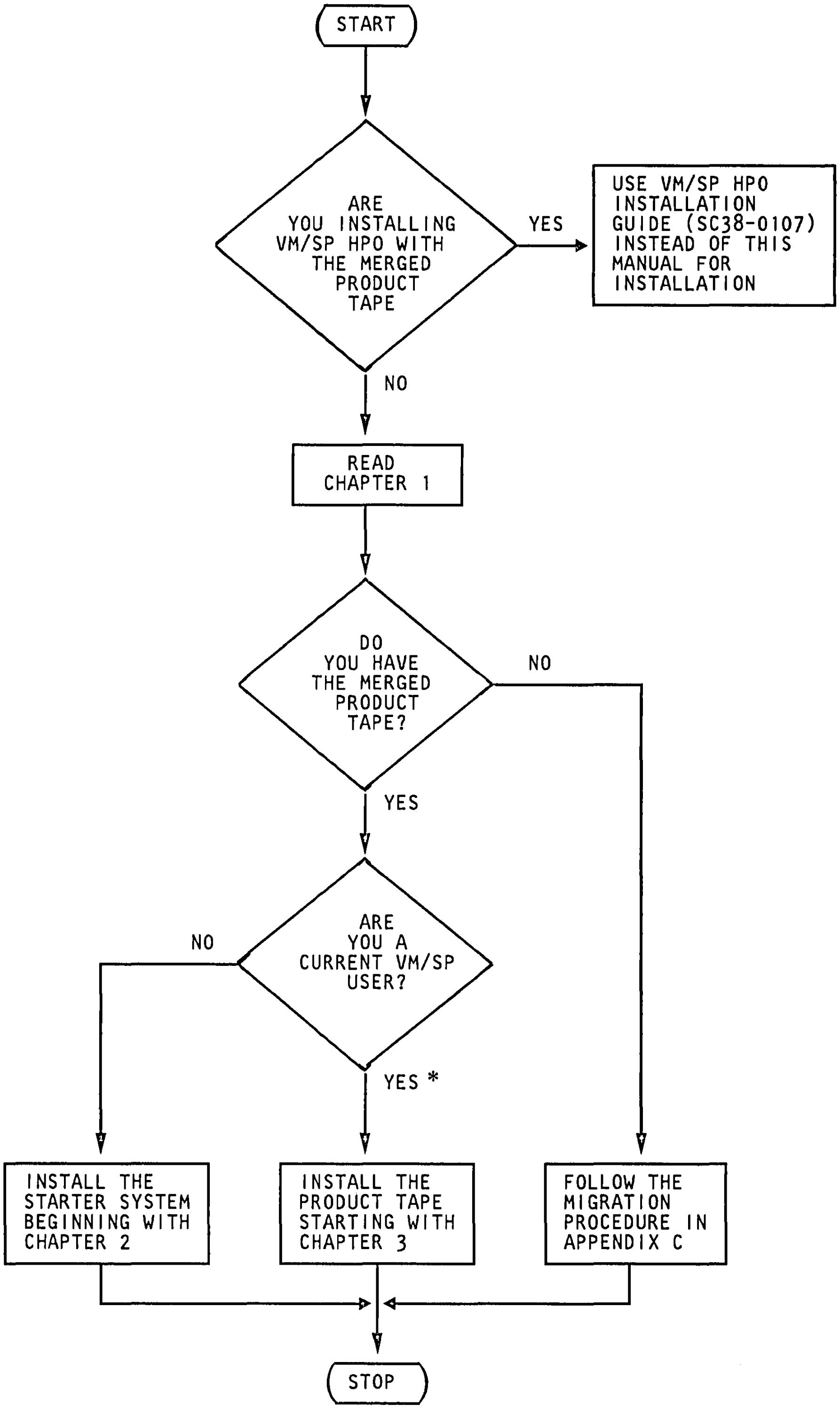
The tape that contains the Optional Feature Program Products that you will install with your system using INSTFPP.

Where to Start

As shown in Figure 2 on page 2, after all users read "Chapter 1. Introduction to VM/SP Installation" on page 5, new users of VM/SP with the Merged Product Tape begin with "Chapter 2. Installation When Using the Starter System Tape" on page 15. Existing VM/SP users with the Merged Product Tape proceed to "Chapter 3. Installation When Not Using the Starter System Tape" on page 77. Other migrating users proceed to Appendix C, "VM/SP Without the Starter System and the Merged Product Tapes" on page 251 for the VM/SP generation procedure.

At this time, you should make a copy of the chart in Appendix H, "Installation Reference Worksheet" on page 285 and fill in the blanks as the needed information becomes available. This volume number, address, label, and device type information will be needed during the installation of VM/SP.

Unless you are a current VM/SP user, you must install the Starter System before you attempt to install the Merged Product Tape. (If you are a current VM/SP user, you are also encouraged to install the Starter System for a much simpler procedure.) See Figure 2 on page 2 for the recommended installation procedure for your situation.


START
ARE YOU INSTALLING VM/SP HPO WITH THE MERGED PRODUCT TAPE
YES
USE VM/SP HPO INSTALLATION GUIDE (SC38-0107) INSTEAD OF THIS MANUAL FOR INSTALLATION
NO
READ CHAPTER 1
DO YOU HAVE THE MERGED PRODUCT TAPE?
NO
YES
ARE YOU A CURRENT VM/SP USER?
NO
YES *
INSTALL THE STARTER SYSTEM BEGINNING WITH CHAPTER 2
INSTALL THE PRODUCT TAPE STARTING WITH CHAPTER 3
FOLLOW THE MIGRATION PROCEDURE IN APPENDIX C
STOP

*If you are a current VM/SP user and do not wish to migrate, you can answer "no" to this question. The "yes" path assumes a higher VM knowledge level.

Figure 2. Organization of VM/SP Installation. Where to start depending on user's system.

Chapter 2. Installation When Using the Starter System Tape

Phase 1. Preinstallation Preparation

Note: If you are installing VM/SP HPO with the merged product tape, use the *VM/SP HPO Installation Guide* for installation instead of this manual.

Before beginning the actual installation process, you should obtain the following:

- VM/SP Starter System Tape
- VM/SP Product Tape
- The number of recommended volumes, for your DASD type, to install VM/SP without any optional feature program products. See the table that follows:

DASD Type	**Volumes if No Source Loaded**	**Volumes if Source Loaded**
3310, 3340	3	5
3330	3	4
3350, 3370, 3375, 3380	2	3

- The recommended number of additional volumes, for your DASD type, to install your configuration of optional feature program products. See Appendix I, "Optional Feature Program Products" on page 287.
- One Tape Drive (or more if possible)
- One display terminal from which to enter the necessary commands
- A copy of the chart in Appendix H, "Installation Reference Worksheet" on page 285. This volume number, address, label, and device type information will be needed during the installation of VM/SP.

Review the default DMKxxx files for your DASD type to decide if you can use the system defaults. The DMKxxx files are on the MAINT 191 minidisk. The DMKxxx file listings are available in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

Before choosing to use the defaults, you must be sure that the I/O configuration is suitable for your installation. If it is, you can take the system defaults when you install. If not, you need to create or modify your own DMKRIO. See "Chapter 19. Preparing the Real I/O Configuration File (DMKRIO)" in the *VM/SP Planning Guide and Reference* for more information.

Use the following criteria to help you decide whether the defaults are to be used.

- Your real DASD addresses must match the addresses given in the default DMKRIO.
- Your real operator's console must match either the console or an alternate console defined in the RIOGEN macro of DMKRIO.
- You must have TAPES and GRAPHICS at addresses defined in the DMKRIO.

If these conditions are met, you will be able to use the default DMKRIO to build your system. Tailoring of DMKRIO can be done after the system is running if you want to add or delete devices. If you do not want the defaults or the criteria were not met, you may create a DMKRIO by altering the default file using the System Product Editor.

If you are going to create a new DMKRIO file, you must know your system and its features very well. This includes all of the devices, addresses, and control units.

In the procedure that follows, you will be required to provide addresses and device types for the new system you are building. If you do not plan to use the available system defaults and/or if you prefer to label or address your devices and volumes differently, record the changes you are planning before you begin.
Appendix H, "Installation Reference Worksheet" on page 285 is provided for your convenience so that you will have a record of your plans at the start of the procedure.

Phase 2. Install the Starter System

Phase 2 Overview

In Phase 2 you will be doing:

Step 1. Load and Run the Format Program

Step 2. Restore the Starter System to Disk

Step 3. Load the CP Nucleus

Step 4. Load the CMS Nucleus

The instructions for completing these steps are on the pages that follow.

To improve readability, some prompts and messages in this procedure are modified slightly from the actual displays. For example, message numbers and timestamps have been removed.

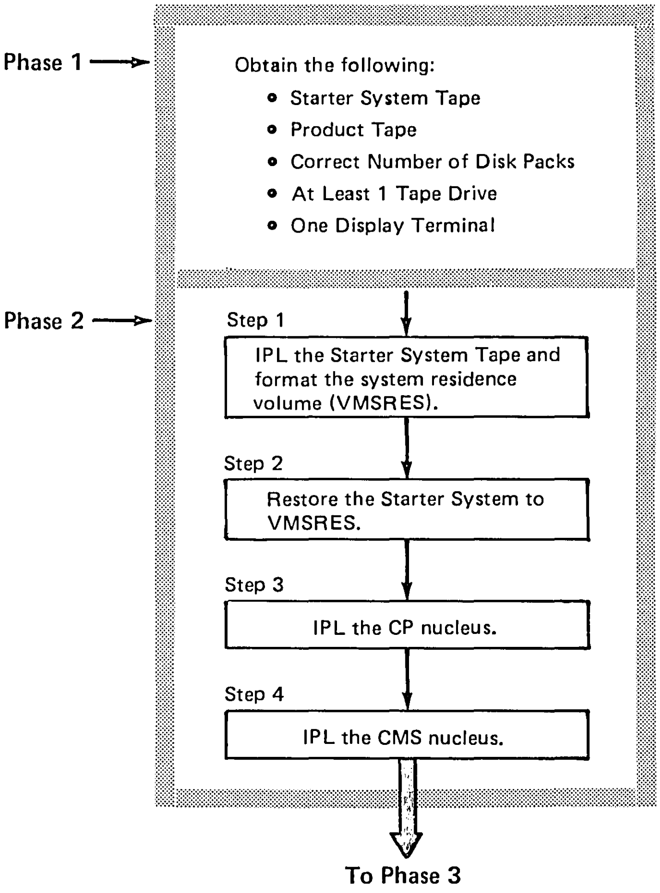


Figure 5. Format VMSRES and IPL Starter System

Step 1. Load and Run the Format Program

If your console address is not 009 or 01F, remember to press the REQUEST key or its equivalent before proceeding. This is necessary to identify it as the system console.

Note: If your console address is something other than 009 or 01F, you need not change the console address on the Program Load Screen.

Figure 6 shows the Starter System Tape format. The FORMAT/ALLOCATE program is the first file on the tape. *Do not rewind the tape after completing this step.*

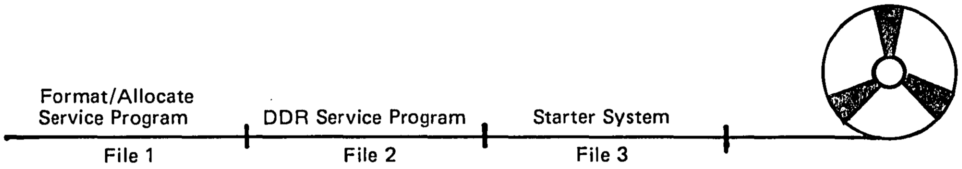


Figure 6. VM/SP Starter System Tape Format

Format and Label the Starter System/System Residence Volume (VMSRES)

The Starter System, when restored, will be the base of the new system residence volume. In this procedure the system residence volume is called VMSRES. VMSRES is formatted in this portion of Step 1.

You will use the FORMAT/ALLOCATE program to format and label the new system residence volume. To execute the FORMAT/ALLOCATE program, respond to the prompting messages at your console.

Note: Formatting large DASD devices can take up to 30 minutes. During this time, the SYSTEM light is on, but there is no screen activity to inform you about the progress of the formatting process. When formatting is done, you will get a message. If, during the formatting process, a problem occurs and you cannot proceed, rewind the Starter System Tape and start over at the beginning of Step 1.

Load the FORMAT/ALLOCATE program from the Starter System Tape in the following manner:

1a. Mount and ready the DASD volumes that are to be used.
1b. Mount and ready the Starter System Tape.
1c. IPL (Initial Program Load) the tape.

- At *first level*, IPL from a tape device according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)

- At *second level*, define VMSRES at virtual address 123 if not previously done and attach a tape drive to your virtual machine.

def cuu 123	cuu is the present virtual address of your VMSRES volume.

Then, mount and IPL the Starter System Tape.

ipl cuu

cuu is the address of the tape drive attached to your virtual machine.

Note: If nothing seems to be happening after IPL, the system is most likely in a wait state. A wait state exists if:

- The SYSTEM light has turned off.
- The WAIT light has turned on.
- The system appears to be inactive.

To initiate the FORMAT/ALLOCATE program from a wait state:

- At *first level*: ENTER
- At *second level*: ATTN (For example, press PA1 on 3278 terminal). Then, ENTER

All standalone programs have limited error recovery routines. If your system fails to respond after the wait state is entered, it could also be a media or tape error. If so, re-IPL the tape.

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:
```

This message indicates that FORMAT/ALLOCATE program is now loaded and can be used.

format

```
FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real disk address on which the new system will be built (VMSRES).

```
ENTER DEVICE TYPE:
```

dt

dt is the device type of the DASD to be formatted. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512. For 3310 or 3370 devices you must enter 'FB-512' when prompted for device type.

```
ENTER START CYLINDER (XXX) OR ''LABEL'':
```

ENTER

The system will display: `ENTER START PAGE (XXX) OR "LABEL":` for FB-512 devices (3310 or 3370). The null entry defaults to 000.

```
ENTER END CYLINDER (XXX):
```

ENTER

The system will display: `ENTER END PAGE (XXX)` for FB-512 devices (3310 or 3370). Based on the device type you previously entered, the null entry results in one of the following values:

(CKD devices)		**(FB-512 devices)**	
3330	end cyl. 403	3310	end page 15751
3330-11	end cyl. 807	3370	end page 69749
3340-35	end cyl. 347		
3340-70	end cyl. 695		
3350	end cyl. 554		
3375	end cyl. 958		
3380	end cyl. 884		

```
ENTER DEVICE LABEL:
```

vmsres

```
FORMAT STARTED

FORMAT DONE
000 NO. PAGE RECORDS WITH READ-CHECK
ERRORS
```

If the previous message indicates other than **000 NO. PAGE RECORDS WITH READ-CHECK ERRORS**, discontinue the installation process and contact your Systems Engineer or hardware service personnel. (You may have to initialize the volume with the Device Support Facilities program. Refer to the *Device Support Facility User's Guide and Reference.*)

Note: Formatting large disks can take up to 30 minutes.

At completion of Step 1 your console displays:

```
ENTER FORMAT OR ALLOCATE:
```

At *first level*, proceed to Step 2.

At *second level* press ATTN then ENTER.

You may want to format additional volumes at this time while the format/allocate program is still operational. For all DASD types, PREP EXEC formats at least two DASD packs. Therefore, volumes that should not be formatted at this time are VMPK01, VMSEXT, and VMSTGE. The PREP EXEC always formats VMPK01 and VMSTGE. For 3310, 3330, and 3340 DASD types it also formats VMSEXT.

Warning: The VMPK01 and VMSEXT volids are contained in the starter system SYSOWN list. Therefore, work volumes must not currently be labeled VMPK01 or VMSEXT if they are to be formatted later by the PREP exec. If they are currently labeled VMPK01 or VMSEXT, then use the FORMAT/ALLOCATE program to relabel these volumes prior to proceeding, or disable these volumes prior to proceeding with Step 3.

Proceed to Step 2.

Step 2. Restore the Starter System to Disk

In this procedure, volume uses are as follows:

VMSRES
Holds the contents of the starter system tape and the product tape until the starter system portion of the installation procedure completes. The starter system then becomes the operational system.

VMPK01
Required for installation using the starter system. The PREP EXEC and GENERATE EXEC use TEMP and T-DISK space allocated on this pack for work space.

VMSEXT
Used for additional work space, or to hold program products if a 3310, 3330, or 3340 is used as the SYSRES device. This pack is used but not required for the installation procedure.

VMSTGE
Used to hold source code. If your installation does not require source code to be maintained on DASD, this pack is not required.

Sample allocations for VMSRES, VMPK01, VMSEXT, and VMSTGE can be found in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

The text on this page was deleted.

The text on this page was deleted.

Step 2

Load the DASD DUMP RESTORE program (second tape file) from the Starter System Tape.

- If you are running *first level*, IPL your tape drive according to the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)

- If you are running *second level*, enter:

ipl cuu

cuu is the address of the tape drive attached to your virtual machine.

Note: You may have to IPL the tape drive twice to get past the tape mark. Also, you may need to press the ENTER key when the system goes into a wait state in order to initiate the DASD DUMP/RESTORE program.

Restore the VMSRES volume by responding as follows to the DASD DUMP/RESTORE program prompts:

```
VM/370 DASD DUMP/RESTORE PROGRAM - VM/SP
ENTER CARD READER ADDRESS OR CONTROL STATEMENTS
ENTER:
```

sysprint cons

```
ENTER:
```

input cuu nnnn

cuu is the address of the tape drive containing the starter system tape.
nnnn is the type of the tape drive: 2400, 2420, 3420, 3430, or 8809. (Specify 3410 as 3420.)

```
ENTER:
```

output cuu dt vmsres

cuu is the address of the disk drive on which the Starter System is to be restored.
dt is the device type of the DASD to be restored. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512 for 3310 or 3370.

```
ENTER:
```

restore all

```
RESTORING VMSRES
DATA DUMPED MM/DD/YY AT HH.MM.SS GMT FROM VMSRES RESTORED TO VMSRES
INPUT CYLINDER EXTENTS       OUTPUT CYLINDER EXTENTS
        START     STOP       START     STOP
        ....      ....       ....      ....
        ....      ....       ....      ....
        ....      ....       ....      ....
END OF RESTORE
```

The start and stop cylinder/block extents will vary, depending on the Starter System device type. Also, the block extents may not be displayed if the console is in DISPLAY mode.

```
ENTER:

  ENTER

END OF JOB
```

If you need to restart the Starter System restore process, do the following:

1. Ready the Starter System Tape.
2. IPL the Starter System tape drive to bypass the FORMAT/ALLOCATE program.
3. Start over at the beginning of Step 2 on page 21.

Otherwise, proceed to the next page.

If you are restoring to a 3310, read this page; otherwise, skip this page.

3310 Reallocation

If you are restoring a 3310 Starter System, complete this section. *Otherwise*, proceed to the next page.

- Rewind and ready the Starter System Tape.
- IPL the Starter System Tape.

Reply to the following prompts:

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:
```

allocate

```
ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real disk address on which the Starter System was just restored.

```
ENTER DEVICE TYPE:
```

fb-512

```
ENTER DEVICE LABEL:
```

vmsres

```
ENTER ALLOCATION DATA FOR VOLUME VMSRES
TYPE PAGE PAGE
```

Forces allocation table to be rewritten.

perm 2 11559

```
ENTER:
```

end

```
ALLOCATION RESULTS
PERM         XXXXX   XXXXX
DRCT         XXXXX   XXXXX
PERM         XXXXX   XXXXX
TEMP         XXXXX   XXXXX
PERM         XXXXX   XXXXX
DEVICE CUU VOLUME VMSRES    ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Continue with Step 3 on page 28 to IPL the Starter System and define the devices required for the system installation.

If you have a 3330-11 device type, read this page; otherwise, go to Step 3.

3330-11 Reallocation

If you are restoring a 3330-11 Starter System, complete this section. *Otherwise*, proceed to the next page.

The portion of the disk space from cylinder 404 to cylinder 807 on the VMSRES volume should be re-allocated for minidisk use.

- Rewind and ready the Starter System Tape.
- IPL the Starter System Tape.

Reply to the following prompts:

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:

  allocate

ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):

  cuu
```

cuu is the real disk address on which the Starter System was just restored.

```
ENTER DEVICE TYPE:

  3330-11

ENTER DEVICE LABEL:

  vmsres

ENTER ALLOCATION DATA FOR VOLUME VMSRES
TYPE CYL CYL
.... ... ...

 perm 404 807
```

Then press ENTER.

```
 end

ALLOCATION RESULTS
PERM        XXXXX   XXXXX
DRCT        XXXXX   XXXXX
PERM        XXXXX   XXXXX
TEMP        XXXXX   XXXXX
PERM        XXXXX   XXXXX
DEVICE CUU VOLUME VMSRES   ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Continue with Step 3 on the next page to IPL the Starter System and define the devices required for the system installation.

Step 3. Load the CP Nucleus

General Considerations Before IPLing the Starter System

If you have control units that share more than 16 devices, and are also switchable to a second processor (non-IPL), care needs to be taken. While you perform the system generation, put the channel-interface-enable switch on the non-IPL processor in the DISABLE position. Any other loosely-coupled processor (via channels) should be placed in STOP mode during actual IPL of the system.

Note: An IPL in a second level environment takes longer than you might expect. The symptom appears as a performance degradation or hung condition. It is caused by contention for service by the devices on the shared control units. Also, if the missing interrupt handler (MIH) is turned on, message MSDMKDID546I (INTERRUPTION CLEARED) appears repeatedly. This is not an indication of a problem; the system is operating correctly.

Warning: If you plan to execute PREP CPFMT, disable all volumes already labeled *VMSEXT and/or VMPK01*. To do this:

- **At *first level*, set the appropriate channel ENABLE/DISABLE switch or switches on your control unit to the DISABLE position.**
- **At *second level*, define the volumes in Step 3 at virtual addresses other than those you chose in Steps 1 and 2. Later in the process, these volumes will be redefined and attached as required.**

If you are operating at *first level*, proceed to page 31 to IPL and define your starter system. If at *second level*, continue below.

Considerations Regarding Second Level Operation

The formatting of volumes and minidisks with *second level* operations causes overhead at all levels. Expect extremely long job completion times. It is recommended that these steps *not* be done at *second level*, if possible.

If, however, you choose to operate at *second level*, some additional entries are suggested depending on whether you are operating with local or remote terminals.

- **Running a Second Level Operation with Local Terminals**

  To have full screen System Product Editor support on local devices when operating at *second level*, complete the following procedure.

  Note: All devices from Figure 3 on page 13 and Figure 4 on page 13 must be attached to the userid *after* logon.

logon userid	Any valid directory entry can be used if you are not currently logged on to your *first level* system.
set ecmode on	
def 01f 009	

Warning: The graphics device (secondary console) must not be addressed with the same first two prefix symbols used in the address of the primary console device. For example, the console device at address 01F disallows any graphics address of the form 01x. Thus, all graphics device addresses ranging from 010 to 01F are not usable for console address 01F.

To bypass this problem, you must either redefine the primary console device address to 009 as shown above or use the default address of 020 to define the graphics (secondary) console.

Check your primary console address by issuing:

```
query console
```

Use the following table to decide which address to use for your secondary console.

If primary console is	**Then secondary console is**
01F	**020 or greater**
009	**010 or greater**

For more detail, refer to the cautionary notes under "DEDICATE Control Statement" (Chapter 18) and under "Miscellaneous Restrictions" (Appendix D) in the *VM/SP Planning Guide and Reference.*

Continue by entering the next command.

term conmode 3270[6]	Display unit must be local non-SNA device.
term linend %	Allows the # sign to be used for *second level* linend.
define stor 1m	Minimum; can be larger if needed.

[6] If the message **DMKCFT006E INVALID DEVICE TYPE - nnnn** appears, continue in the procedure for "Remote Terminals" on the next page (just prior to "term linend %").

- **Running a Second Level Operation with Remote Terminals**

  To have full-screen support when executing the starter system from a virtual machine on an existing system using remote terminals, complete the procedure below.

  Note: All devices from Figure 3 on page 13 and Figure 4 on page 13 must be attached to the userid *after* logon.

  Enter these commands from the primary user console:

  ```
  logon userid            Use any valid directory entry.

  set ecmode on
  ```

  This is the entry point for users who had the **INVALID DEVICE TYPE - nnnn** message from the procedure for "Local Terminals."

  ```
  term linend %           Allows the # sign to be used for second level CP
                          linend.

  def graf 020
  ```

  IPL the starter system (VMSRES). This causes the starter CP nucleus to load.

  ```
  ipl cuu clear           cuu is the virtual address of VMSRES.
  ```

  The messages and responses as a result of IPLing will be similar to those that appear on pages 32 through 34. You'll need to respond accordingly before you continue with the commands that follow.

  ```
  vary on 020

  enable 020
  ```

  Once the system is up, log off the current userid MAINT on the master console.

  ```
  dial userid             Same as logon userid above.

  logon maint cpcms       Logging on another physical console as your graphics
                          device using your MAINT userid.
  ```

  This allows full console support on the second device. All running will be done on this device, but the primary console must remain active.

Note: To avoid problems with system security and maintain system integrity, remember to change all system provided passwords when feasible. Referring to the example above, CPCMS is a password that is provided with the system. It (and any related minidisk passwords) should be changed to ones of your own choosing when your system becomes fully operational.

If you have a failure at this time, recover by reIPLing the Starter System.

The following message will appear when you reIPL:

```
*** DO YOU WISH TO RE-DEFINE YOUR SYSTEM *** (YES|NO):
```

Respond **yes** if the device addresses are not specified correctly, or if the tapes and/or disks have been moved. If you reply **yes**, you will be restarting in Step 3 on page 31 (See RESTART HERE). Otherwise, reply **no**.

Continue with Step 4 on the next page.

Step 4. Load the CMS Nucleus

You are now ready to load CMS. The Starter System is running and you are logged on as user MAINT automatically.

At this time your console displays:

```
VM/SP n STARTER
```

ENTER

```
R;
```

The CMS nucleus is now loaded.

Note:

If you previously disabled the volumes labeled VMSEXT and/or VMPK01, and wish to use them as 'work' volumes, enable them now. If you didn't disable the volumes labeled VMSEXT and/or VMPK01, continue with Step 4 on the next page.

To enable the volumes:

- At *first level*, set the appropriate channel ENABLE/DISABLE switch or switches on your control unit to the ENABLE position.
- At *second level*, attach the volumes redefining the volume addresses to those you'entered in reply to the Starter System messages on the previous pages. Then, vary on the *second level* system by entering the following:

  def cuu1 cuua — cuu1 is your work volume address.
  cuua is the work volume address used in Step 3.

  def cuu2 cuub — cuu2 is your extra work volume address.
  cuub is the extra work volume address used in Step 3.

Continue with Phase 3 on the next page.

```
000 NO. PAGE RECORDS WITH READ-CHECK ERRORS
ENTER FORMAT OR ALLOCATE:
```

No reply necessary to this response.

```
ALLOCATE
```

Begin allocating VMPK01.

```
ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
125
ENTER DEVICE TYPE:
3330-11
ENTER DEVICE LABEL:
VMPK01
ENTER ALLOCATION DATA FOR VOLUME VMPK01
TYPE CYL CYL
.... ... ...
PERM 000 119
TDSK 120 214
TEMP 215 334
PERM 335 807
END
ALLOCATION RESULTS
PERM 000 119
TDSK 120 214
TEMP 215 334
PERM 335 807
```

```
DEVICE 125  VOLUME VMPK01 ALLOCATION ENDED
```

VMPK01 allocation complete.

```
ENTER FORMAT OR ALLOCATE:
```

No reply necessary to this response.

Formatting and allocating the work disks are now complete.

Warning: When running a *second level* machine, the *first level* linend character # has to be changed to some other character. Enter TERM LINEND % if you did not do so previously.

To clear old data and make the results of PREP CPFMT allocation known to CP, issue:

- At *first level*, **#cp ipl 190**
- At *second level*, **#cp ipl 190 parm autocr**

Step 6

```
VM/SP n STARTER
```

ENTER

This entry only necessary at *first level.*

```
R;
```

Issue **detach 124** if VMSEXT was just formatted in this step. (124 is a virtual address.) Only needed for 3310, 3330, or 3340 devices.

detach 124

```
DASD ' CUU1 ' DETACHED MAINT 124
R;
```

Issue **detach 125** if VMPK01 was just formatted in this step. (125 is a virtual address.)

detach 125

```
DASD ' CUU1 ' DETACHED MAINT 125
R;
```

attach cuu1 **system vmpk01**

cuu1 is the real device address of your extra work volume, VMPK01.

```
DASD ' CUU1 ' ATTACH TO SYSTEM VMPK01
R;
```

attach cuu2 **system vmsext**

cuu2 is the real device address of your work volume, VMSEXT. Only needed for 3310, 3330, or 3340 devices.

```
DASD ' CUU2 ' ATTACH TO SYSTEM VMSEXT
R;
```

D. PREP DOHELP: Help file management

The PREP DOHELP option allows you to erase HELP files or translate them to uppercase. The HELP files on the CMS system disk extension are in mixed lowercase and uppercase character representation. In some installations, lowercase characters are reserved for display of special alphabets. In such installations, HELP files should be displayed in uppercase representation only.

To initiate this option, enter:

prep dohelp

```
' DOHELP ' PROCESS STARTING ...
ENTER "ERASE" TO ERASE ALL HELP FILES ON THE SPECIFIED DISK.
ENTER "UP"    TO TRANSLATE ALL HELP FILES TO UPPERCASE REPRESENTATION.
ENTER "EXIT"  TO EXIT.
```

up

```
THE HELP FILES TO BE TRANSLATED TO UPPER CASE ARE
ASSUMED TO RESIDE ON DISK 19D.  IF THEY RESIDE ELSEWHERE,
ENTER DISK (E.G. 191, 192, 19C. ETC.).
PRESS ENTER FOR DEFAULT. TYPE EXIT TO EXIT.
```

ENTER

```
'19D' REPLACES ' B (190) '
R;
```

If you have moved the CMS system disk to a disk other than the one designated by the Starter System, you can translate the HELP files to uppercase character representation on your new minidisk and keep the Starter System as it is.

Proceed to Step 8 on the next page.

Step 8. Apply Service Updates

Note: Any PUT tape *Memo to Users* information replaces or supplements information in this step. Be sure to review it before applying service.

A VM/SP service tape might be supplied with VM/SP. This tape contains cumulative service for VM/SP. You will now use the two service minidisks, 293 and 294, that you formatted in Step 6(B) using PREP CMSFMT.

Mount the service tape (with no ring) as virtual drive 181 and issue the following commands:

```
attach cuu * 181
```

cuu is the real address of the service tape device.

```
access 191 c
vmfplc2 load * * c
vmserv
```

The VMSERV EXEC maps the service tape and allows you to print the PUT document and *Memo to Users.*

```
DO YOU WANT TO PRINT THE MEMO TO USERS?
yes
```

VMSERV then prints the PUT document and *Memo to Users*, and reminds you to read them prior to installing service. After reviewing the documentation and contacting IBM concerning the latest service activity, you can install service as described in the PUT document and *Memo to Users.*

Do *not* build a new CP or CMS nucleus as described in the user memo, but continue here after applying service from the PUT. After applying service, recreate the ASSEMBLE module as follows:

```
access 190 a
asmgend
ipl 190 parm autocr
```

Proceed to Phase 5 on the next page.

Step 9. Modify System Installation-Dependent Files

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied with VM/SP serve as a base for building files unique to your installation. You may need to modify one or more of these files now.

Examine the listings of the Sample Directory and DMK*xxx* files which are reproduced in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

Use the System Product Editor to modify your sample files if desired. For more information, see "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference.*

Directory Considerations

Note: The optional feature program product execs (PPPREP and INSTFPP) assume that certain disks exist with certain passwords. You should not delete or alter passwords of minidisks related to optional feature program products that you are installing until after they are installed.

The VM/SP SAMPLE directory is the VMUSERS DIRECT file loaded in Step 6.

If you are installing optional feature program products, you may need to modify the directory (VMUSERS DIRECT); for example, due to space requirements. Directory related information for each of the optional feature program products is shown in Appendix I, "Optional Feature Program Products" on page 287. You can update this file later to include additional virtual machine definitions and userids.

Prior to modifying the directory:

1. Refer to Figure 26 on page 290 to determine the size 319 minidisk you need.

2. Relocate any minidisk, if necessary, to create space for expanding MAINT's 319 minidisk.

3. Verify that all minidisks required for the program products you are installing are located on the DASD volumes you have. If they are not, move them as needed.

See "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference* for a general discussion. For more detailed information, see "A Virtual Machine for Updating VM/SP" on page 142.

If you made changes to the directory, you may want to use the DISKMAP EXEC to check for erroneous overlaps.

DMKRIO Considerations

DMKRIO defines the real I/O configurations. If the default DMKRIO file does not match your system installation, you need to alter it now, before you build your new CP nucleus. A copy of the DMKRIO file can be found in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

For more information about modifying your own DMKRIO, see "Chapter 19. Preparing the Real I/O Configuration File (DMKRIO)" in the *VM/SP Planning Guide and Reference.*

DMKSNT Considerations

If you plan to add products, other than optional feature program products that require a saved segment, the SNT entries should be added now. This prevents later reassembly of DMKSNT and CP nucleus regeneration. Check user-selected product installation documentation for SNT related information. A description of CMS saved segment relocation can be found in "Planning for CMS" in Part 1 of the *VM/SP Planning Guide and Reference.*

For the procedure regarding alternate CMS nucleus placement, see Appendix E, "Example of Alternate CMS Nucleus Placement" on page 275. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for an example to modify the CMS nucleus to include file status tables for the S- and Y-minidisks.

DMKSYS Considerations

No space dedicated to paging has been provided in the system samples. An additional volume is required for paging space. The new volume id should be added to the SYSOWN macro definition to contain the paging space. Refer to "Direct Access Storage Requirements for CP" in the *VM/SP Planning Guide and Reference* for more information.

During execution of the GENERATE EXEC process, you will be given the opportunity to write the CP nucleus to tape. Be aware that the SYSRES=123 definition in the sample DMKSYS file requires a real address of 123, if this was to be used as a standalone backup tape. If you wish to change this to a real address, you must also change the corresponding DIRECT MDISK 123 statement for the MAINT userid and the DIRECTORY 123 statement in the sample directory.

Other Considerations

The following files can be altered as required to meet your installation's needs.

- Printer Universal Character Set[10]

  DMKUCS (1403)
  DMKUCB (3211)
  DMKUCC (3203)
  DMKPIB (3262)

- Font Offset Buffer[10]

  DMKPIA (3289)

- Forms Control Macros[10]

  DMKFCB (forms control)

- LOGO

  DMKBOX

Proceed to Step 10.

[10] See "Updating Printer Modules" on page 192 for further information.

This page intentionally left blank.

Step 10. Generating a CP Nucleus

The procedure for generating a CP nucleus is managed by GENERATE EXEC. GENERATE EXEC provides entry points for generating selected portions of the CP nucleus. Figure 10 on page 54 shows the entry points and exits of the sequence.

An entry point is selected by specifying the appropriate operand following the GENERATE command. For instance, placing the TEMP operand after GENERATE causes a complete CP nucleus to be created.

Note: If you plan to build the CP nucleus using GENERATE CP NUCLEUS instead of GENERATE TEMP, you must assemble all of the DMKRIO, DMKSYS, and DMKSNT files prior to entering the command.

Four of the operands allow an ONLY option. If "ONLY" is specified after the DIRECT, DMKRIO, DMKSYS, or DMKSNT operand, *only* the GENERATE code associated with the specific operand is executed.

If you would like to display HELP information that describes any GENERATE entry point, enter:

```
generate help
```

For more detailed information on the GENERATE EXEC, see "GENERATE" on page 197.

If you are using one tape drive only, issue:

def 181 182 Allows you the option to save the CP nucleus on tape later in this step.

Replace the tape currently mounted with a scratch tape (with ring) to receive the nucleus code.

If you plan to include the Small CP or Virtual=Real option, read Appendix G, "Special Options for CP" on page 283 before proceeding.

This procedure assumes that you enter GENERATE EXEC from the top. To start, get temporary disk space by entering:

generate temp

The TEMP option causes execution of the entire GENERATE EXEC. The GENERATE TEMP option formats and allocates temporary minidisk 192 (B-disk) for assembler workspace. If an error occurs, the routine exits at the step being processed.

```
GET TEMP SPACE ON DASD TYPE = 3330
DISK 'B' NOT ACCESSED
DEV 192 DOES NOT EXIST
DASD 192 DEFINED
FORMAT - TEMP 30 CYLINDER MDISK.
TEMP MDISK 192 ACCESSED AS FILE MODE B.

DO YOU WISH TO UPDATE THE SYSTEM
DIRECTORY AND ASSEMBLE THE SYSTEM
DEFINITION FILES NOW?
RESPOND ( YES | NO )
```

Executing GENERATE TEMP code.

yes

yes means that RIO, SNT, and SYS will be assembled and the directory file will be placed online.
no results in only a temporary minidisk being allocated and an exit from GENERATE EXEC.
Executing GENERATE VMSP code.

```
NO GENERATE DEFAULTS FILE EXISTS
STANDARD VM/SP DEFAULTS WILL BE USED
'194' REPLACES ' A (191) '
C (191) R/O
```

```
PRT FILE nnnn TO MAINT COPY 001 NOHOLD
ENTER DIRECTORY FILENAME:
THE NULL DEFAULT VALUE IS 'VMUSERS'.
TO BYPASS STEP ENTER 'SKIP' -
(GO TO DMKRIO ASSEMBLE STEP)
```

Executing GENERATE DIRECT code.

ENTER

VMUSERS is now your directory filename.

```
EOJ DIRECTORY UPDATED AND ON LINE

WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:    ( NO | YES )
```

This sample procedure assumes a uniprocessor system.

no

```
VMFASM BEING EXECUTED WITH
FN= DMKRIO CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKRIO

ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKRIO TEXT A1 CREATED
PRT FILE nnnn TO MAINT COPY 001 NOHOLD
```

Executing GENERATE DMKRIO code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSYS CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSYS

ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSYS TEXT A1 CREATED
PRT FILE nnnn TO MAINT COPY 001 NOHOLD
```

Executing GENERATE DMKSYS code.

```
VMFASM BEING EXECUTED WITH
FN= DMKSNT CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKSNT

ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKSNT TEXT A1 CREATED
PRT FILE nnnn TO MAINT COPY 001 NOHOLD
```

Executing GENERATE DMKSNT code.

Before you go on, the GENERATE TEMP command and the functions it performs must complete successfully. If any errors occurred while the directory was being built or while the DMKxxx files were assembling, the directory program and the GENERATE EXEC procedure issued the message:

```
EDIT AND CORRECT THE DIRECTORY FILE IN ERROR.

                      - or -

EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR.
```

If you received one of these messages, read the next page for the procedure to correct the error condition. Otherwise, continue on page 61.

Read this page if you received one of the error messages on the previous page.

ERROR CONDITIONS

A. "EDIT AND CORRECT THE DIRECTORY FILE IN ERROR" Message

Errors were detected while the Directory was being loaded.

The DIRECT Option reads the directory file from your minidisk and writes it to the system residence volume. Specifying GENERATE DIRECT ONLY causes an exit from the GENERATE routine after the new directory is written. If "ONLY" is not specified, the routine continues with assembly of DMKRIO, DMKSYS, and DMKSNT. You need to edit and correct the Directory file using the System Product Editor. Then, respond with:

generate direct

B. "EDIT AND CORRECT THE 'filename' ASSEMBLE FILE IN ERROR" Message

Errors were detected during assembly of one or more of the DMKRIO, DMKSYS, or DMKSNT files.

Use the System Product Editor to correct the indicated "filename" (DMKRIO, DMKSYS, or DMKSNT). Then, enter the GENERATE command with the appropriate option. See the following example:

generate dmkxxx

dmkxxx represents **dmkrio, dmksys, or dmksnt.**

```
WILL YOU BE GENERATING A 'MULTI-
PROCESSOR' (MP) SYSTEM?
ENTER:   ( NO | YES )
```

This message appears only when generating DMKRIO.

no

This sample procedure assumes a uniprocessor system.

```
VMFASM BEING EXECUTED WITH
FN= DMKxxx CNTRL= DMKSP
NO UPDATE FILES WERE FOUND.
ASMBLING DMKxxx
ASSEMBLER (XF) DONE
NO STATEMENTS FLAGGED IN THIS ASSEMBLY
DMKxxx TEXT CREATED
```

- If **dmkxxx** = **dmkrio**, then GENERATE DMKRIO is executed. The DMKRIO Operand assembles DMKRIO. If the ONLY option is not specified, it also:

  1. Assembles DMKSYS and DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksys**, then GENERATE DMKSYS is executed. The DMKSYS Operand assembles DMKSYS. If the ONLY option is not specified, it also:

  1. Assembles DMKSNT.
  2. Generates a CP nucleus.

- If **dmkxxx** = **dmksnt**, then GENERATE DMKSNT is executed. The DMKSNT Operand assembles DMKSNT. If the ONLY option is not specified, it also generates a CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete.

```
DO YOU WANT THE SMALL CP OPTION? --
ENTER:   ( NO | YES )
```

See Appendix G, "Special Options for CP" on page 283 if you want the Small CP option.

```
   no
```

Executing GENERATE CP NUCLEUS. The CP NUCLEUS operand causes the CP nucleus to be generated. A CP nucleus can be written to tape or to the target disk. If the NOLOAD option is specified, the resulting nucleus will not be IPLed automatically.

```
WHICH ARE YOU GENERATING: A
'UNI PROCESSOR' (UP),
AN 'ATTACHED PROCESSOR' (AP),
OR A 'MULTI PROCESSOR' (MP) SYSTEM ?

ENTER: ( UP | AP | MP )

   up
```

Your responses to these questions determine the CNTRL file and LOADLIST EXEC used to correctly build your system.

```
VIRTUAL=REAL OPTION REQUIRED (YES|NO):
```

See Appendix G, "Special Options for CP" on page 283 if you want the V=R option.

```
   no

NO FILES PURGED

VMFLOAD BEING EXECUTED
LOADLIST= CPLOAD AND CNTRL=DMKSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT  COPY 001   NOHOLD

AN IPL'ABLE CP NUCLEUS EXISTS IN YOUR VIRTUAL CARD READER.
DO YOU WISH TO CREATE AN IPL'ABLE TAPE
OR, LOAD THE NUCLEUS IN YOUR READER DIRECTLY
TO YOUR SYSTEM RESIDENCE DISK VOLUME?
ENTER: ( DISK | TAPE | EXIT ).

   tape
```

You are saving the new CP nucleus on tape as well as on DASD. If any errors are detected while the tape is being written, you must recreate the CP nucleus. To do this, enter GENERATE CP NUCLEUS. The procedure then restarts at the point above where you are asked: `DO YOU WANT THE SMALL CP OPTION? --`

```
IPLABLE NUCLEUS NOW ON TAPE ****
```

Note: This tape is not sufficient for a disaster recovery.

```
WHEN YOU RECEIVE THE MESSAGE: NUCLEUS LOADED ON 'volid'
-ENTER: 'IPL 190 PARM AUTOCR'
TO PUT THE CP LOAD MAP ONTO YOUR 194 MINIDISK, AFTER THE IPL
-ENTER: GENERATE MAP

NUCLEUS LOADED ON VMSRES - STARTING CYL/BLK=397,
LAST CYL/BLK USED=401
```

The previous message gives the starting and ending cylinders/blocks for the nucleus. Verify that this nucleus has not expanded into another allocated area such as "TEMP," "WARM START," etc., or unpredictable results may occur.

The message below does not automatically display on many processors. Check your processor reference manual for information about displaying and interpreting the PSW. If the PSW message is not displayed, press PA1 to get a CP READ state, then proceed.

```
CP ENTERED; DISABLED WAIT
PSW '0002 0000 0000 0012'
```

This is the normal return if the nucleus loaded correctly.

If you do not receive the **NUCLEUS LOADED ON 'volid'** message, read the following procedure to correct the error condition. Otherwise, continue on page 63.

ERROR CONDITION

No "NUCLEUS LOADED ON 'volid' " Message

- Inspect the error load map in the virtual reader and the virtual PSW. A loader error may be indicated on the listing or PSW. See the *VM/SP System Messages and Codes* for a list of the loader wait state codes.

  After correcting the error, issue GENERATE CP NUCLEUS.

- If you shut down following a nucleus loading error, you may not be able to reIPL the system. If so:

  1. Mount the Starter System Tape if it is not already attached.
  2. IPL the Starter System Tape *twice* to load the DDR program.
  3. Follow Step 2 of this procedure, starting on page 21, through the first three commands that define input and output devices. Then, enter RESTORE commands, with parameters appropriate for your DASD type, as follows:

     For FB-512 (3310/3370) devices:

restore 00000 00015	Restore Starter System allocation.
restore 92480 94303	Restore Starter System directory.
restore 123736 126015	Restore Starter System nucleus.

     For CKD devices:

restore 000 000	Restore Starter System allocation.
restore aaa bbb	Restore Starter System directory.
restore xxx yyy	Restore Starter System nucleus.

     Where aaa bbb and xxx yyy are:

Device Type	aaa	bbb	xxx	yyy
3330	275	278	397	403
3340	070	079	032	043
3350	101	102	549	554
3375	144	147	953	958
3380	097	098	881	884

  You can now reIPL, recreate your directory using GENERATE DIRECT ONLY, and generate a new CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete. To reload CMS, enter:

ipl 190 parm autocr

For *second level* operation, ENTER after IPLing.

```
VM/SP n STARTER
R;
```

generate map

Executing GENERATE MAP to load the CP nucleus map onto the 194 minidisk.

```
LOAD MAP HAS BEEN PLACED IN READER


SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cp

```
'194' REPLACES ' A (191) '
RECORD LENGTH IS '132' BYTES
YOU HAVE NOW SAVED "CPNUC MAP" ON MDISK "194"

R;
```

The contents of the load map are described in "Chapter 9. Updating An Installed VM/SP System" on page 165. Two external names may be listed as undefined on the load map. The external name DMKSLC is undefined if the Virtual=Real option is not selected. The external name DMKRNTBL is undefined if there is no entry in the system name table for a 3704/3705 control program. (Provided you did not code a NAMENCP macro in the DMKSNT file.) Also, other names might be listed as undefined if other modules were deleted as described in "Reducing the CP Nucleus Size" in Part 1 of the *VM/SP Planning Guide and Reference.*

You have now generated the CP nucleus having had the chance to select Small CP or V=R options. At this point continue with Step 11.

Step 11. Generating a CMS Nucleus

To initiate this step, enter:

generate cms nucleus

The exact text of the first message issued by GENERATE CMS NUCLEUS depends on the device type being used for the CMS system disk (S-disk).

```
NO GENERATE DEFAULTS FILE EXISTS
STANDARD VM/SP DEFAULTS WILL BE USED
' 190 ' REPLACES ' A(194) '
A (190) R/O

3330 IS THE "S DISK" DASD TYPE.
IF SAMPLE SIZES ARE USED, REPLY TO THE PROMPT "CYL|BLK NUMBER":
WITH ' 94 ' IF THE CMS NUCLEUS IS TO BE GENERATED OR
WITH ' 97 ' IF THE CMSL NUCLEUS IS TO BE GENERATED
NO FILES PURGED


VMFLOAD BEING EXECUTED
LOADLIST= CMSLOAD AND CNTRL= DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn  TO  MAINT COPY 001   NOHOLD

SAVE A COPY OF THE CMS NUCLEUS AS A DISK FILE?
ENTER: ( NO | YES )
```

yes

```
THE CMS NUCLEUS NOW EXISTS ON DISK AS FILE ' CMSNUC NUCLEUS A1 '.
TO SAVE THE MAP FROM THIS NUCLEUS GENERATION,

ENTER: GENERATE MAP
```

Ignore this message until after CMS nucleus is re-IPLed on page 65.

```
CMS LOAD MESSAGES APPEAR NEXT.
RESPOND AS INDICATED.
```

See "Chapter 9. Updating An Installed VM/SP System" on page 165 for an explanation of the prompts and responses.

```
DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

This device address is the virtual address where the CMS nucleus is to be written.

190

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =

   94
```

The nucleus will reside on the last cylinder(s) or block(s) of the 190 minidisk. Your S-disk DASD type is displayed in the first message in Step 11. (3330 in this sample.) Respond with the cylinder/block number also displayed in that message. (**94** in this sample.) Use the accompanying chart for the values you need for your device type.

Device	CMS	CMSL
FB-512	43520 blk	44544 blk
3330	094 cyl	097 cyl
3340	222 cyl	229 cyl
3350	045 cyl	047 cyl
3375	070 cyl	072 cyl
3380	043 cyl	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?

   yes

DMSINI611R VERSION IDENTIFICATION =

   ENTER
```

You are permitted to enter up to 32 descriptive characters to identify the version and level of CMS. This information is displayed each time you IPL the CMS system now being generated.

```
DMSINI612R INSTALLATION HEADING =

   ENTER
```

You are permitted to enter up to 64 descriptive characters to serve as an installation heading at the beginning of each output file.

```
VM/SP REL n  mm/dd/yy hh:mm

   ENTER
```

The new CMS nucleus is automatically reIPLed by **ENTER**

```
R;

   generate map

LOAD MAP HAS BEEN PLACED IN READER


SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END

   cms

'191' REPLACES ' A (191) '
RECORD LENGTH IS '132' BYTES
YOU HAVE NOW SAVED "CMSNUC MAP" ON MDISK "191"
R;
```

Proceed to Phase 6 on the next page.

Phase 6. Conclude the Sample System Installation

Phase Overview

In Phase 6 you will be doing:

Step 12. IPL the newly generated system.

Step 13. Format the operator's 191 minidisk.

Step 14. Updating the SYSTEM NETID.

Step 15. Save CMS.

Step 16. Create a backup system on tape (optional).

The instructions for completing these steps are on the pages that follow.

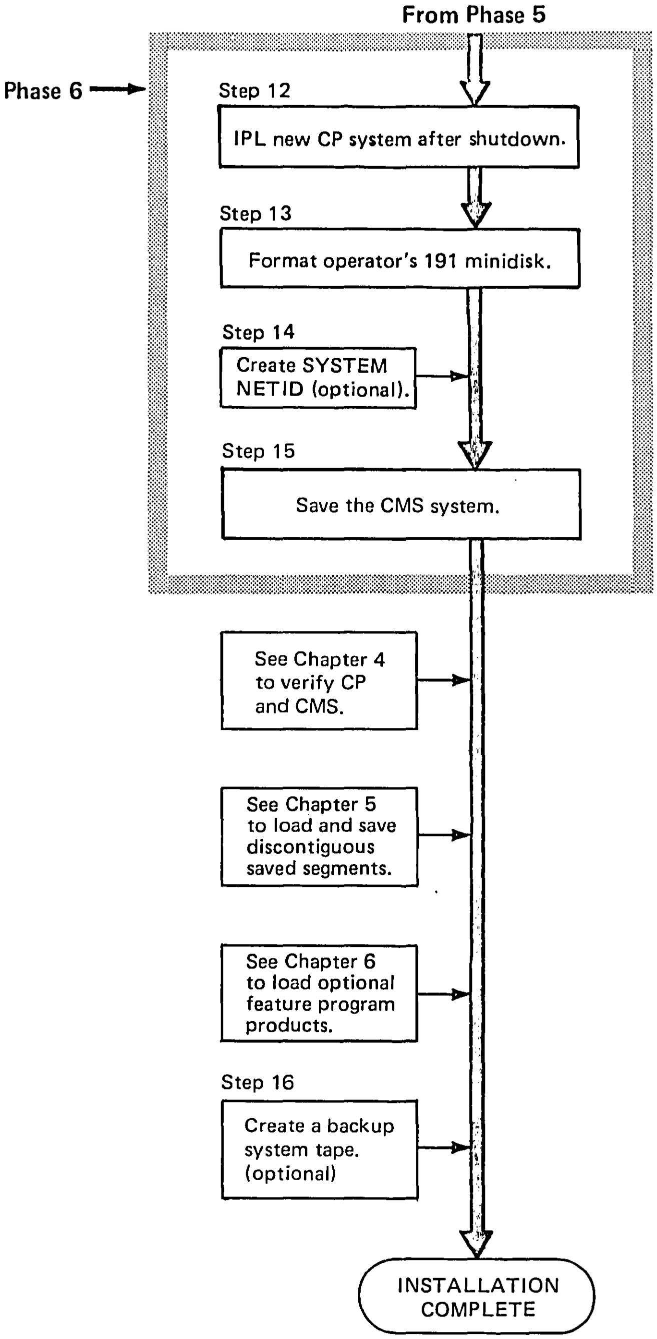

From Phase 5
Phase 6
Step 12
IPL new CP system after shutdown.
Step 13
Format operator's 191 minidisk.
Step 14
Create SYSTEM NETID (optional).
Step 15
Save the CMS system.
See Chapter 4 to verify CP and CMS.
See Chapter 5 to load and save discontiguous saved segments.
See Chapter 6 to load optional feature program products.
Step 16
Create a backup system tape. (optional)
INSTALLATION COMPLETE

Figure 11. Conclude Sample System Installation

Step 12. IPL the Newly Generated System

In this step, you IPL your generated VM/SP system rather than the starter system.

Be aware that the output is spooled class T. If you want the output to be printed, you must alter the class of the output or the printer.

First, spool your console closed by issuing:

```
  sp con close
R;
```

Second, drain all spooling devices by entering:

```
  drain all
R;
```

Be sure to wait until all attached I/O devices are drained as indicated by the **R;** ready message.

Then, shutdown the Starter System by entering:

```
  shutdown
```

The system operator userid, assigned in DMKSYS (SYSOPR macro), is logged on when you IPL the newly created system residence volume.

- At *first level*, IPL (Initial Program Load) the new system using the real address of VMSRES. Follow the directions for your machine type. (For specific information on the IPL process for your machine, see the *Operator's Guide* provided with your hardware.)
- At *second level*, define VMSRES as follows:

  def cuu xxx — cuu is the present virtual address of your VMSRES volumes. xxx matches the DMKRIO entry for that DASD type (of the VMSRES pack).

  Then, IPL the virtual address of VMSRES.

  ipl 123 — **123** for this sample.

```
VM/SP REL n, SERVICE LEVEL 0000

NOW hh:mm:ss EDT day mm/dd/yy
CHANGE TOD CLOCK (YES|NO):
```

At *first level*, refer to your hardware documentation. At *second level*, always reply **no**.

```
  no
START ((COLD|WARM|CKPT|FORCE)
(DRAIN))|(SHUTDOWN):
```

Because there is no data or accounting information to be recovered, you can request a cold start.

```
  cold
```

```
FORMATTING DISK 'A'.

'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

'nn' is the number of cylinders assigned to 191 in the directory.

- If you do not have a second terminal available, logoff. Then, log back on using the MAINT id.

  logoff Enter only if no second terminal available.

  logon maint cpcms

- If you have a second terminal available:

  – At the operator console:

    enable all

  – At the 2nd terminal:

    logon maint cpcms

Note: To avoid problems with system security and maintain system integrity, remember to change all system provided passwords when feasible. Referring to the example above, CPCMS is a password that is provided with the system. It (and any related minidisk passwords) should be changed to ones of your own choosing when your system becomes fully operational.

Proceed to Step 14.

Step 14. Updating the SYSTEM NETID File for CMS Productivity Aids (Optional)

If you have an RSCS Networking virtual machine, you should update a file on the CMS 190 disk called SYSTEM NETID. The SYSTEM NETID file is referenced when you issue CMS commands to communicate via the network. The CMS IDENTIFY command is used by the NOTE, SENDFILE, and TELL commands to transmit notes, files, and messages.

The IDENTIFY command uses the CP QUERY CPUID command to retrieve the CPU serial number. This number is then looked up in the SYSTEM NETID file. IDENTIFY uses the CP QUERY USERID command to retrieve the node. If there is a conflict in nodes between the SYSTEM NETID file and CP QUERY USERID, the node in SYSTEM NETID takes precedence. If there is no record with a matching serial number, or if the SYSTEM NETID file is not found, the 'rscsid' is set to an asterisk (*).

The logon of the MAINT userid causes an automatic IPL 190.

```
VM/SP REL n  MM/DD/YY HH:MM
   ENTER
```

Completes the automatic IPL.

```
R;
   term mode vm
```

This command allows CMS commands to be issued successfully.

```
   set emsg on
R;
```

Provides message numbers for messages displayed in this step.

```
   acc 190 b
R;
```

Running from the MAINT userid allows you to write to the 190 minidisk where the SYSTEM NETID file will be stored.

If you plan to use the TELL, NOTE, or SENDFILE command, you should update the supplied file. The records in this file have the format:

```
ssssss nodeid rscsid
```

where 'ssssss' is your CPU serial number, 'nodeid' is your local node, and 'rscsid' is the userid of the RSCS virtual machine.

To obtain your CPU serial number, issue:

```
    q cpuid
CPUID = FFssssssdddd0000
R;
```

'ssssss' is your CPU serial number and 'dddd' is your processor device type.

```
    xedit system netid b2
    file
R;
```

Update the SYSTEM NETID file with your CPU serial number (ssssss), local node (nodeid corresponding to the local nodeid of the RSCS disconnected virtual machine as it appears in the RSCS directory found on its 191 disk), and userid of the RSCS virtual machine (rscsid as it appears in the CP directory).

Proceed to Step 15.

This page intentionally left blank.

Step 15. Save CMS

By using the DMKSNT layout and the new CP nucleus, you can now save your CMS system.

The default virtual storage size when running under the MAINT userid is 6M. That is the required storage for this step.

Note: In the procedure that follows, be sure to enter the SAVESYS CMS command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMS system, enter:

```
  ipl 190 clear
VM/SP REL n  MM/DD/YY HH:MM
```

The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

```
  savesys cms
```

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

```
 ENTER
SYSTEM SAVED
VM/SP RELEASE n  MM/DD/YY HH:MM
 ENTER
```

Your CMS system is now saved. You can issue IPL CMS instead of IPL 190, when you wish to run CMS.

Step 16. Create a Backup Tape (Optional)

This step is optional if the product will not be installed or if segments will not be saved.

> **Note:** If you did not apply service to your new VM/SP system and did not load the source files, you only have to backup your VMSRES; however, if you have done one of those things, you may have to do more work:
>
> - If you applied service to your Release 3.0 system and you installed it on 3310s, 3330s, or 3340s, you also need to create a backup of your VMPK01 volume because that is where the minidisks for MAINT 293 and 294 are located.
> - All users who loaded the CP and CMS source files will have to backup their VMSTGE volume, while users who installed their system on 3310s or 3340s will also have to create a backup of their second source volume VMSTG2.
>
> This assumes that the default minidisk locations are not changed.

The DASD Dump Restore (DDR) program is described in the *VM/SP Operator's Guide*. Before using the DDR control statements to back up your system residence volume (VMSRES), insure VMSRES is located at virtual address 123 by issuing:

```
q v dasd
```

Load and ready a scratch tape. Then, attach the scratch tape as your backup by issuing:

```
att cuu maint 181
```

cuu is the real address of your tape drive.

Then, continue the backup procedure by entering:

```
   ddr

ENTER:

   input 123 3330 vmsres

ENTER:

   output 181 dt (mode 6250

ENTER:

   dump cpvol
```

dt represents the type of tape drive. **dump cpvol** causes cylinder 0 and any disk cylinders allocated as PERM or DRCT to be dumped onto the tape.

```
DMKDDR711R VOLID READ IS VMSRES
DO YOU WISH TO CONTINUE?  RESPOND YES, NO OR REREAD:

   yes

ENTER NEXT EXTENT OR NULL LINE

   ENTER

DUMPING   VMSRES
END OF DUMP

   ENTER

END OF JOB
PRT FILE nnnn  FOR VMSP3  COPY 001  HOLD
R;
```

You have completed the VM/SP Installation procedure. Verify that your new CP and CMS are working properly by turning to page 101 for the Installation Verification Procedure.

Step 2. Load the Product Tape and Build a New CMS Nucleus.

CAUTION

For users migrating from VM/SP Release 1, PREP EXEC will not execute properly unless it is running from the minidisk accessed as c.

To load the Product Tape, issue:

```
prep load define nuc
```

This results in prompts for addresses of minidisks that will receive the new product code. A CMS nucleus is then created and loaded at the new product level.

Note: Much of the manual generation that was required for installation in Release 1 has been eliminated by the PREP EXEC. PREP EXEC automates the steps previously needed to generate VM/SP. For further insight into exactly what PREP EXEC does, print a copy of it and follow the generation procedure in more detail.

```
' LOAD ' PROCESS HAS BEEN REQUESTED:

'LOAD' PROCESS STARTING.
NO RDR FILES.
YOU HAVE CHOSEN THE OPTION TO BUILD A
CMS NUCLEUS AFTER LOAD.
CAUTION: THIS EXEC PURGES CLASS I READER
FILES.  CONTINUE? (Y OR N)

   y

CAUTION: NO ERROR CHECKING IS
DONE FOR MINIDISK AVAILABILITY,
THE MDISKS TO BE ENTERED IN THE
FOLLOWING RESPONSES MUST BE CURRENTLY
ATTACHED TO YOUR VIRTUAL MACHINE.

SINCE YOU DO NOT WISH TO USE THE
DEFAULT MINIDISKS TO LOAD THE PRODUCT
TAPE, RESPOND TO THE FOLLOWING PROMPTS.

ENTER WHERE CP FILES ARE TO
BE LOADED ( CUU | FSF )
```

FSF = Forward Space File

```
   194
```

194 is the address of the CP text minidisk.

```
ENTER WHERE HELP FILES ARE TO
BE LOADED ( CUU | FSF )
```

19d

19d is the address of HELP minidisk.

```
ENTER WHERE CMS FILES ARE TO
BE LOADED ( CUU | FSF )
```

490

```
THE ATTACHED MDISKS ARE DEFINED AS FOLLOWS:
  CP MDISK = 194
  HELP MDISK = 19D
  CMS MDISK = 490
ARE THESE RESPONSES CORRECT? ( Y | N | EXIT )
```

y

```
' LOAD ' PROCESS STARTING . . .

LOADING   MAINT 194 MINIDISK (CP TEXT DECK DISK)
MAINT 194  LOADED

LOADING   MAINT 19D MINIDISK (HELP FILE DISK)
MAINT 19D  LOADED

LOADING MAINT 490 MINIDISK (CMS TEXT FILE DISK)
MAINT 490 LOADED

THE PRODUCT TAPE HAS BEEN LOADED TO DISK. IF SOURCE IS
NOT TO BE LOADED TO DISK, YOU MAY WISH TO REPLACE
THE PRODUCT TAPE WITH A SCRATCH TAPE AT THIS POINT.
'LOAD' EXECUTION COMPLETE.

GENERATING THE SYSTEM ASSEMBLER COMMAND.
NO RESPONSE IS REQUIRED FOR THE ASSEMBLE PROMPTS.
ASSEMBLE XF GEND PROC
ENTER TARGET DISK MODE FOR ASSEMBLE MODULES.
DEFAULTS TO S-DISK IF NONE ENTERED.
ASSEMBLE XF GEND COMPLETE.
SYSTEM LOAD DECK COMPLETE.
DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =

  490
```

The response to this prompt is the address of the device where the CMS nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =

  94
```

The response to this prompt is device dependent. See page 82 for the correct value for your device type.

```
DMSINI610R ALSO IPL CYL/BLK 0 ?

  yes
```

```
DMSINI611R VERSION IDENTIFICATION =

  ENTER
```

You are permitted to enter up to 32 descriptive characters to identify the version and level of CMS. This information is displayed each time you IPL the CMS system now being generated.

```
DMSINI612R INSTALLATION HEADING =

  ENTER
```

You are permitted to enter up to 64 descriptive characters to serve as an installation heading at the beginning of each output file.

```
VM/SP REL n MM/DD/YY
DMSACC723I Y (19E) R/O[12]
R;
```

Warning: Do not issue "SAVESYS CMS" at this point.

In order to use the "new" CMS, define the 190 minidisk as 590. Then, define the 490 minidisk (the "new" CMS) as 190 in your virtual machine.

def 190 590

Redefines the "old" CMS to 590.

def 490 190

Redefines the "new" CMS to 190. Necessary so that the "new" CMS can be used in place of the "old" CMS.

IPL the new CMS by issuing:

ipl 190

You *must* IPL by using the "new" CMS minidisk address since you do not wish to load the "old" CMS system.

To verify that the new system is operational, execute some basic CP and CMS commands, for example: QUERY DISK and QUERY DASD.

> **Note:** You can use the GENERATE MAP function at this time to load the map.

Continue with Step 3.

[12] This message is displayed only when the Y-STAT is too large to fit in the CMS nucleus. If desired, the CMS nucleus can be generated to allow space for a large Y-STAT. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for additional information.

Step 3. Perform Optional PREP EXEC Tasks

During the installation process you might want to alter your existing files or tailor some sample files shipped with the system. You may also want to load system source code or load utility programs. There are PREP EXEC options to assist you. They are:

A. PREP SAMPLE - Print out copies of sample files.
B. PREP LOADSRC - Load CP and/or CMS source code.
C. PREP UTIL - Create stand-alone utility program tape.
D. PREP DOHELP - Erase HELP files or convert them to uppercase.

To initiate any PREP EXEC task, enter the PREP command with the appropriate operand, for example: PREP UTIL. If you need more information about a specific command, you can enter the HELP option following the operand. For example, if you want to know more about PREP UTIL, you enter PREP UTIL HELP.

For a more detailed sample, see pages 48 through 51 in "Chapter 2. Installation When Using the Starter System Tape" on page 15.

Continue with Step 4.

Step 4. Apply Service Updates

Note: Any PUT tape *Memo to Users* information replaces or supplements information in this step. Be sure to review it before applying service.

A VM/SP service tape may be supplied with VM/SP. This tape contains cumulative service for VM/SP.

Mount and attach the service tape (with no ring) as virtual drive 181. Now issue the following commands:

```
attach cuu * 181
```

cuu is the real address of the service tape device.

```
access 191 c

vmfplc2 load * * c

vmserv
```

The VMSERV EXEC maps the service tape and allows you to print the PUT document and *Memo to Users.* Reply **yes** to the next prompt.

```
DO YOU WANT TO PRINT THE MEMO TO USERS?

    yes
```

VMSERV then prints the PUT document and *Memo to Users*, and reminds you to read them prior to installing service. After reviewing the documentation and contacting IBM concerning the latest service activity, you can install service as described in the PUT document and *Memo to Users.*

Do *not* build a new CP nucleus as described in the user memo, but continue here after applying service from the PUT. After applying service, recreate the ASSEMBLE module as follows:

```
access 190 a

asmgend

ipl 190 parm autocr
```

Continue with Step 5.

Step 5. Modify System Installation-Dependent Files

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied with VM/SP serve as a base for building files unique to your installation. You may need to modify one or more of these files now.

The sample DIRECT, DMKRIO, DMKSNT, and DMKSYS files supplied in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

are intended to serve as a base for building files unique to your installation. Examine the listings of the Sample Directory and DMK*xxx* files and use the System Product Editor to modify these sample files if desired. For more information, see "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference.*

Directory Considerations

Note: The optional feature program product execs (PPPREP and INSTFPP) assume that certain disks exist with certain passwords. You should not delete or alter passwords of minidisks related to optional feature program products that you are installing until after they are installed.

If you are ready to build the CP nucleus, your directory (VMUSERS DIRECT) will need to be modified now, reflecting the changes you defined in Step 2. Namely, you need to change MAINT's 190 MDISK to 590 and MAINT's 490 MDISK to 190 before proceeding. You can update this file later to include additional virtual machine definitions and userids.

Prior to modifying the directory:

1. Refer to Figure 26 on page 290 to determine the size 319 minidisk you need.
2. Relocate any minidisk, if necessary, to create space for expanding MAINT's 319 minidisk.
3. Verify that all minidisks required for the program products you are installing are located on the DASD volumes you have. If they are not, move them as needed.

See "Part 2. Defining Your VM/SP System" in the *VM/SP Planning Guide and Reference* for a general discussion. For more detailed information, see "A Virtual Machine for Updating VM/SP" on page 142.

DMKRIO Considerations

Your current DMKRIO file probably has not changed unless you have altered I/O devices with this installation. Make any necessary changes to the DMKRIO files at this point before you build the new CP nucleus.

DMKSNT Considerations

Because CMS and CMSL segments have been modified, you may want to merge the new sample DMKSNT with your existing file. Check user-selected product installation documentation for system name table related information.

A description of CMS saved segment relocation can be found in "Planning for CMS" in Part 1 of the *VM/SP Planning Guide and Reference.* For the procedure regarding alternate CMS nucleus placement, see Appendix E, "Example of Alternate CMS Nucleus Placement" on page 275. See Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for an example to modify the CMS nucleus to include file status tables for the S- and Y-minidisks.

DMKSYS Considerations

No space dedicated to paging has been provided in the system samples. An additional volume is required for paging space. The new volume id should be added to the SYSOWN macro definition to contain the paging space. Refer to "Direct Access Storage Requirements for CP" in the *VM/SP Planning Guide and Reference* for more information.

If you do not receive the **NUCLEUS LOADED ON 'volid'** message, read the following procedure to correct the error condition. Otherwise, continue on page 96.

ERROR CONDITION

No "NUCLEUS LOADED ON 'volid'" Message

- Inspect the error load map in the virtual reader and the virtual PSW. A loader error may be indicated on the listing or PSW. See the *VM/SP System Messages and Codes* for a list of the loader wait state codes.

  After correcting the error, issue GENERATE CP NUCLEUS.

- If you shut down following a nucleus loading error, you may not be able to reIPL the system. If so:

  1. Mount the tape created by the GENERATE command in the initial generation procedure.

  2. IPL that tape.

  3. Recreate your directory using GENERATE DIRECT ONLY, and generate a new CP nucleus.

The procedure continues on the next page.

CP nucleus build is complete. To reload CMS, enter:

```
   ipl 190 parm autocr
```

For *second level* operation, ENTER after IPLing.

```
VM/SP n STARTER
R;

   generate map

LOAD MAP HAS BEEN PLACED IN READER
```

Executing GENERATE MAP to load the CP nucleus map onto the 194 minidisk.

```
SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END

   cp

'194' REPLACES ' A (194) '
YOU HAVE NOW SAVED "CPNUC MAP" ON MDISK "194"

R;
```

The contents of the load map are described in "Chapter 9. Updating An Installed VM/SP System" on page 165. Two external names may be listed as undefined on the load map. The external name DMKSLC is undefined if the Virtual=Real option is not specified. The external name DMKRNTBL is undefined if there is no entry in the system name table for a 3704/3705 control program (Provided you did not code a NAMENCP macro in the DMKSNT file). Also, other names might be listed as undefined if other modules were deleted as described in "Reducing the CP Nucleus Size" in Part 1 of the *VM/SP Planning Guide and Reference.*

You have now generated the CP nucleus having had the chance to select Small CP or V=R options. To run under the new CP nucleus you just generated, you must shutdown the system and re-IPL. Be sure to save any spool files before shutting down the system.

Use the SPTAPE command to save spool files. For details on the SPTAPE command, refer to the *VM/SP Operator's Guide.* At this point continue with Step 7.

Step 7. Updating the SYSTEM NETID File for CMS Productivity Aids (Optional)

If you have an RSCS Networking virtual machine, you should update a file on the CMS 190 disk called SYSTEM NETID. The SYSTEM NETID file is referenced when you issue CMS commands to communicate via the network. The CMS IDENTIFY command is used by the NOTE, SENDFILE, and TELL commands to transmit notes, files, and messages.

The IDENTIFY command uses the CP QUERY CPUID command to retrieve the CPU serial number. This number is then looked up in the SYSTEM NETID file. IDENTIFY uses the CP QUERY USERID command to retrieve the node. If there is a conflict in nodes between the SYSTEM NETID file and CP QUERY USERID, the node in SYSTEM NETID takes precedence. If there is no record with a matching serial number, or if the SYSTEM NETID file is not found, the 'rscsid' is set to an asterisk (*).

The logon of the MAINT userid causes an automatic IPL 190.

```
VM/SP REL n  MM/DD/YY HH:MM
  ENTER
```

Completes the automatic IPL.

```
R;
  term mode vm
```

This command allows CMS commands to be issued successfully.

```
  set emsg on
R;
```

Provides message numbers for messages displayed in this step.

```
  acc 190 b
R;
```

Running from the MAINT userid allows you to write to the 190 minidisk where the SYSTEM NETID file will be stored.

If you plan to use the TELL, NOTE, or SENDFILE command, you should update the supplied file. The records in this file have the format:

```
ssssss nodeid rscsid
```

where 'ssssss' is your CPU serial number, 'nodeid' is your local node, and 'rscsid' is the userid of the RSCS virtual machine.

To obtain your CPU serial number, issue:

```
  q cpuid
CPUID = FFssssssdddd0000
R;
```

'ssssss' is your CPU serial number and 'dddd' is your processor device type.

```
  xedit system netid b2
  file
R;
```

Update the SYSTEM NETID file with your CPU serial number (ssssss), local node (nodeid corresponding to the local nodeid of the RSCS disconnected virtual machine as it appears in the RSCS directory found on its 191 disk), and userid of the RSCS virtual machine (rscsid as it appears in the CP directory).

Proceed to Step 8.

Step 8. Save CMS

By using the DMKSNT layout and the new CP nucleus, you can now save your CMS system.

The default virtual storage size when running under the MAINT userid is 6M. That is the required storage for this step.

Note: In the procedure that follows, be sure to enter the SAVESYS CMS command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMS system, enter:

ipl 190 clear

```
VM/SP REL n   MM/DD/YY HH:MM
```

The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

savesys cms

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide*.

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP RELEASE n   MM/DD/YY HH:MM
```

Your CMS system is now saved. You can issue IPL CMS instead of IPL 190, when you wish to run CMS.

Note: EXEC procedures under CMS that help you load, set storage keys for, and save CMS discontiguous saved segments are:

EXEC	**DSS Named Segment**
DOSGEN	CMSDOS and INSTVSAM
VSAMGEN	CMSVSAM and/or CMSAMS
SAMGEN	CMSBAM
GENERATE	CMSL

The information you need to complete this task is contained in "Chapter 5. Loading and Saving Discontiguous Saved Segments" on page 109. Other essential material is located in "Generating Saved Systems" in the *VM/SP System Programmer's Guide*.

From the system operator's console, enter:

```
  ipl cms

  term mode vm
```

Then, initiate the IVP tests by entering:

```
  ivp
```

Next, answer the following question:

```
ARE YOU THE SYSTEM OPERATOR?
ENTER ''YES'' OR ''NO'':

  yes (or no)
```

A **no** reply defaults to the single virtual machine verification procedure.

Prompting instructions are displayed whenever you must perform an operation or issue a command.

From this point use a second terminal device (one other than the operator's console device).

Log on the virtual machine (IVPM1), using the password IVPASS, and IPL CMS to continue the testing procedure:

```
  logon ivpm1

ENTER PASSWORD:

  ivpass

LOGON AT hh:mm:ss EST FRIDAY mm/dd/yy

  def stor 16m

STORAGE=16384K
```

Note: To avoid problems with system security and maintain system integrity, remember to change all system provided passwords when feasible. Referring to the example above, IVPASS is a password that is provided with the system. It (and any related minidisk passwords) should be changed to ones of your own choosing when your system becomes fully operational.

IPL CMS by issuing:

```
  ipl 190
```

The system will respond:

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

If the IVPM1 191 minidisk is not already formatted, enter:

```
  acc (nodisk
```

The NODISK option prevents CMS from automatically accessing your virtual disk 191. (Accessing 191 at this time would cause an error message to be issued because 191 is not yet initialized, and therefore cannot be used.)

After the ready message is displayed, issue:

```
  format 191 a
```

The CMS FORMAT command prompts you with:

```
FORMAT WILL ERASE ALL FILES ON DISK 'A(191)'.
DO YOU WISH TO CONTINUE? (YES|NO):

  yes

ENTER DISK LABEL:

  ivp1
```

Enter the one-to-six character alphameric label of the virtual disk. You can use whatever label you wish for this virtual disk.

```
FORMATTING DISK 'A'.
'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

Message issued by CMS.

Then you begin the IVP procedure by issuing:

```
  ivp 1
```

At this point, the tests begin on virtual machine 1. After the disconnect message is displayed, follow the prompting messages that are displayed. These messages tell you to log on the IVPM2 virtual machine (this may be done on the same terminal), as follows:

```
logon ivpm2

ivpass
```

```
LOGON AT hh:mm:ss EST FRIDAY mm/dd/yy
```

Note: To avoid problems with system security and maintain system integrity, remember to change all system provided passwords when feasible. Referring to the example above, IVPASS is a password that is provided with the system. It (and any related minidisk passwords) should be changed to ones of your own choosing when your system becomes fully operational.

IPL CMS by issuing:

```
ipl 190
```

The system will respond:

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

If the IVPM2 191 minidisk is not already formatted, enter:

```
acc (nodisk
```

The NODISK option prevents CMS from automatically accessing your virtual disk 191. (Accessing 191 at this time would cause an error message to be issued because 191 is not yet initialized, and therefore cannot be used.)

After the ready message is displayed, issue:

```
format 191 a
```

The CMS FORMAT command prompts you with:

```
FORMAT WILL ERASE ALL FILES ON DISK 'A(191)'.
DO YOU WISH TO CONTINUE? (YES|NO):
```

yes

```
ENTER DISK LABEL:
```

ivp2

Enter the one- to six-character alphameric label of the virtual disk. You can use whatever label you wish for this virtual disk.

CMS then issues:

```
FORMATTING DISK 'A'.
'nn' CYLINDERS FORMATTED ON 'A(191)'.
R;
```

Then you begin the IVP procedure by issuing:

ivp 2

At this point, the remainder of the tests begin on virtual machine 2. The final phase of the IVP tests consists of displaying, printing, and punching a file that contains the messages generated by IVPM1 after it is disconnected.

Upon completion of the tests, the IVP EXEC procedure logs off. The system abnormal termination test, which consists of forcing an ABEND dump of VM/SP and the subsequent warm start, is an option that you may specify in response to messages that are displayed. For the purpose of installation verification, you should select this option. You are instructed to delay starting the spooling devices (reader, printer, and punch) until after the warm start procedure.

Interpreting the Test Results

Messages at the end of the IVP test indicate successful completion. If any errors are detected by the IVP, call IBM for software support because an error usually indicates a serious malfunction of the generated system. The IVP procedure identifies each command being tested just before the command is executed.

Error messages are displayed in a four-line format, for example:

```
*** IVP FAILURE HAS OCCURRED ***
*** COMMAND: STATE IVPTST *
*** EXPECTED RETURN CODE 28
*** RECEIVED RETURN CODE 0
```

These messages indicate that the CMS STATE command had a return code of 0 instead of the expected 28.

All information messages that originate within the IVP are preceded by three asterisks (***).

If any command fails, the IVP procedure terminates. Follow the instructions (if any are given) to log off the virtual machine.

Once the IVP procedure has executed successfully, log on your new system as MAINT. Issue:

```
logon maint

cpcms
```

Note: To avoid problems with system security and maintain system integrity, remember to change all system provided passwords when feasible. Referring to the example above, CPCMS is a password that is provided with the system. It (and any related minidisk passwords) should be changed to ones of your own choosing when your system becomes fully operational.

Then, continue the system installation process by loading and saving discontiguous saved segments, as described in "Chapter 5. Loading and Saving Discontiguous Saved Segments" on page 109.

Chapter 5. Loading and Saving Discontiguous Saved Segments

Information Regarding Discontiguous Saved Segments

The DMKSNT ASSEMBLE file supplied with the Product Tape includes entries for saved segments called CMS, CMSL, CMSVSAM, CMSAMS, CMSDOS, CMSBAM, and INSTVSAM in addition to the saved segments required by the optional feature program products. See "Preparing the System Name Table File (DMKSNT)" in Part 2 of the *VM/SP Planning Guide and Reference* for more information.

Throughout the following discussion, it will be helpful for you to refer to Figure 13 on page 111, which shows where the CMS discontiguous saved segments are loaded in virtual storage using the suggested system name table (DMKSNT) file.

Before a discontiguous saved segment can be attached and detached by name, it must be loaded and saved. It must be loaded at an address that is beyond the highest address of any virtual machine that will attach it. Make sure the saved segment is loaded at an address that does not overlay the defined virtual machine or any other saved segment that may be attached at the same time. Determine the load addresses by checking the entries coded in your DMKSNT module.

> The load address for the discontiguous saved segment should be slightly higher than the highest storage address of the virtual machine using it. If the load address is unnecessarily high, real storage is wasted because CP must have segment table entries for unused storage.
>
> For example, assume you have five CMS virtual machines in your installation. Also assume that all five use the CMS support for VSE program development and testing which is in a 64K segment named CMSDOS. If each of your five CMS virtual machines has a machine size of 320K, you should load the CMSDOS segment just beyond 320K but below 976K (so as to contain it within 1M). Otherwise, real storage would be wasted because CP must maintain segment table entries for each 1024K (1M) of storage.

Once the named segment is loaded at the correct address, it can be saved. Make sure that a discontiguous saved segment has storage protection. Set the storage key for the segment with the SETKEY command. The CMS SETKEY command is described in the *VM/SP System Programmer's Guide.*

CMS has EXEC procedures that help you load, set storage keys, and save the CMS discontiguous saved segments. These procedures are as follows:

- DOSGEN EXEC which loads and saves CMSDOS and INSTVSAM segments.
- VSAMGEN EXEC which loads and saves CMSVSAM and/or CMSAMS segments.
- SAMGEN EXEC which loads and saves the CMSBAM segment.
- GENERATE EXEC which loads the CMSL nucleus.

Note: All of the procedures used for loading and saving discontiguous saved segments are filemode number 2. The associated text files are filemode number 1 to reduce the amount of storage needed for the master file directory in the user's virtual machine. The system disk must be accessed as any mode other than S, before loading any discontiguous saved segment. Make sure that this is done *after* any intermediate IPL of CMS.

You may want to compare this storage layout with the suggested DMKSNT layout for your DASD type shown in:

Appendix J, "3310 Based System Sample Files" on page 303
Appendix K, "3330 Based System Sample File" on page 333
Appendix L, "3340 Based System Sample Files" on page 363
Appendix M, "3350 Based System Sample Files" on page 395
Appendix N, "3370 Based System Sample File" on page 425
Appendix O, "3375 Based System Sample File" on page 455
Appendix P, "3380 Based System Sample File" on page 487

Note that as new releases of VSE/VSAM become available, they may require more segments, thus requiring all segment addresses above these segments to be increased accordingly.

Note: Refer to the latest VM/SP *Memo to Users* for any changes to the sample layout of CMS storage for discontiguous saved segments.

Relationship of Page Numbers, Segment Numbers, and Hexadecimal Addresses

Since the NAMESYS macro requires you to specify page and segment numbers, and the DOSGEN, SAMGEN, and VSAMGEN procedures require you to enter hexadecimal addresses, you may find the following reference information useful.

1K	=	1024	= X'400'			
4K	=	4096	= X'1000'	=	1 page	
64K	=	65536	= X'10000'	=	16 pages	= 1 segment

To convert a page number to a segment number, divide the page number by 16.

Since one segment is 10000 in hexadecimal, then 20000 is segment 2, 100000 is segment 16, 180000 is segment 24, and so on.

The list of functions you can use for installing VSAM and/or Access Method Services is displayed:

```
SELECT ONE OF THE FOLLOWING FUNCTIONS BY ENTERING THE NUMBER:

 1. INSTALL AMS            (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 2. INSTALL VSAM           (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 3. INSTALL VSAM AND AMS   (READ VSAM PRODUCT TAPE, BUILD DOSLIB, CREATE SEGMENT)
 4. BUILD  AMS             (BUILD DOSLIB, CREATE SEGMENT)
 5. BUILD  VSAM            (BUILD DOSLIB, CREATE SEGMENT)
 6. BUILD  VSAM AND AMS    (BUILD DOSLIB, CREATE SEGMENT)
 7. RESTART AMS            (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
 8. RESTART VSAM           (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
 9. RESTART VSAM AND AMS   (READ OPTIONAL FEATURE PRODUCT TAPE, CREATE SEGMENT)
10. QUIT                   (EXIT VSAMGEN EXECUTION)

       ENTER RESPONSE...
```

While VSAMGEN is processing, you receive error, information, and response type messages. The error and information messages are self-explanatory. Error messages labeled 2101I are information messages from the linkage editor and should be ignored. If you are following the sample DMKSNT shown in Figure 13 on page 111 , respond as follows to the prompting messages.

```
DMSVGN363R ENTER LOCATION WHERE CMSVSAM WILL
           BE LOADED AND SAVED
```

210000

For CMSVSAM, the address must correspond to where DMKSNT expects to load CMSVSAM. **210000** using the sample DMKSNT as shown in Figure 13 on page 111 .

```
DMSVGN366R ENTER NAME OF SYSTEM TO BE SAVED
```

cmsvsam

SAMGEN saves the segment.

```
SGN365I SYSTEM CMSVSAM SAVED
```

Indicates that CMSVSAM has been successfully saved.

```
DMSVGN363R ENTER LOCATION WHERE CMSAMS WILL
           BE LOADED AND SAVED:
```

280000

For CMSAMS the address must correspond to where DMKSNT expects to load CMSAMS. **280000** using the sample DMKSNT as shown in Figure 13 on page 111 .

```
DMSVGN366R ENTER NAME OF SYSTEM TO BE SAVED:
```

cmsams

SAMGEN saves the segment.

```
SGN365I SYSTEM CMSAMS SAVED
```

Indicates that CMSAMS has been successfully saved.

If you wish to save DOSLIB and AMS, you must rename and copy them now.

Continue with Step 4.

Step 4. Load and save CMSDOS

To load and save the CMS/DOS segment called CMSDOS, use the same EXEC procedure (DOSGEN) that you used to load and save the CMS/DOS segment called INSTVSAM in Step 1 on page 112.

If you plan to load the 64K CMS/DOS segment called CMSDOS at 3136K (310000), as in the example shown in Figure 13 on page 111, your virtual machine size must be at least 3328K. Access the CMS system disk as any mode other than S.

Issue the commands:

```
  def stor 4m

  ipl 190

  acc (noprof

R;

  acc 190 b/a dms* text

R;

  dosgen 310000 cmsdos                    Saves the CMS/DOS segment.

R;
```

DOSGEN checks that the address contains valid characters, is greater than X‘20000’, and less than 16M. If an error is detected, the message

```
DMSGEN095E INVALID ADDRESS 'address'
```

is issued and the command is terminated.

DOSGEN then checks for a read/write A-disk on which to write the CMS loader work file; if none is accessed, it issues an error message and terminates.

```
DMSGEN006E NO READ/WRITE A-DISK ACCESSED
```

Next, DOSGEN loads all the text files needed for VSE simulation. The text files are loaded starting at the address specified on the DOSGEN command. If there are any unresolved external references, DOSGEN terminates execution with the message:

```
DMSGEN111E DOSGEN FAILED DUE TO LOAD ERRORS
```

DOSGEN then assigns a storage key of X‘D’ to the segment and saves it. If an error is detected, one of the following messages is issued and DOSGEN terminates execution:

```
DMSGEN412S DOSGEN FAILED DUE TO SETKEY ERRORS
DMSGEN141S DOSGEN FAILED DUE TO SAVESYS ERRORS
```

Otherwise, the segment is successfully saved and the load map is written to the A-disk as LOAD MAP A5.

```
SYSTEM SAVED
DMSGEN715I DOSGEN COMPLETE
```

The system name entry in the SYSNAMES table defaults to LOAD.

If you wish to save the load map, access 190 as B and make a copy of the load map now. For example: copy load map a5 cmsdos segmap b2.

At this point you have completed the procedures for loading and saving the discontiguous saved segments that are defined in the DMKSNT module supplied with the Starter System. If you wish to load and save other discontiguous saved segments, you must have created other DMKSNT entries for them, and you now must repeat these procedures for those entries.

Step 5. Generate and Save the CMSL Nucleus Saved Segment

Assuming you are still logged on using the MAINT userid, you can increase your virtual machine storage size, up to the maximum size defined for it in the VM/SP directory, by entering the DEFINE STORAGE command. Then, reload CMS. Issue:

def stor 16m

ipl 190

```
VM/SP RELEASE n  mm/dd/yy hh:mm
```

Completes the IPL.

generate cmsl nucleus

The system response will be similar to the following:

```
NO GENERATE DEFAULTS FILE EXISTS
STANDARD VM/SP DEFAULTS WILL BE USED
DMSACC724I '190' REPLACES ' A (191) '
DMSACC723I A (190) R/O
DMSACC725I 190 ALSO = S-DISK

3330 IS THE "S DISK" DASD TYPE.
IF SAMPLE SIZES ARE USED, REPLY TO THE PROMPT "CYL|BLK NUMBER":
WITH ' 94 ' IF THE CMS NUCLEUS IS TO BE GENERATED OR
WITH ' 97 ' IF THE CMSL NUCLEUS IS TO BE GENERATED
 NO  FILES PURGED

VMFLOAD BEING EXECUTED
LOADLIST = CMSLOADL AND CNTRL = DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD

TO SAVE THE MAP FROM THIS NUCLEUS GENERATION,
ENTER: GENERATE MAP

 CMS LOAD MESSAGES APPEAR NEXT, RESPOND AS INDICATED.

 NOTE: THE CMSL NUCLEUS OPTION HAS BEEN SELECTED.
 REPLY - NO TO THE QUESTION  'ALSO IPL CYL/BLK 0?'
 ****

DMSINI606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

190

The response to this prompt is the address of the device where the CMSL nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =

  97
```

The response to this prompt is device dependent and only applies if the release 3.1 CMS sample minidisk sizes were used. See the table below for the correct CMSL value for your device type.

Device	CMSL
FB-512	44544 blk
3330	097 cyl
3340	229 cyl
3350	047 cyl
3375	072 cyl
3380	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?

  no

DMSINI611R VERSION IDENTIFICATION =

  ENTER
```

You are permitted to enter up to 32 descriptive characters to identify the version and level of CMS. This information is displayed each time you IPL the CMS system now being generated.

```
DMSINI612R INSTALLATION HEADING =

  ENTER
```

You are permitted to enter up to 64 descriptive characters to serve as an installation heading at the beginning of each output file.

```
VM/SPn xxxxxxxx mm/dd/yy

  ENTER

R;

  generate map

LOAD MAP HAS BEEN PLACED IN READER

SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END

  cmsl

'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSLNUC MAP" ON MDISK "191"
R;
```

Note: In the procedure that follows, be sure to enter the SAVESYS CMSL command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMSL system, enter:

```
  ipl 190 97 clear
```

97 is the cylinder number you used in response to message number DMSINI609R. The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

```
savesys cmsl
```

If the CMSL system is named something other than CMSL, see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP RELEASE n  mm/dd/yy hh:mm
```

Your CMSL system is now saved. You can issue IPL CMSL instead of IPL 190, when you wish to run CMSL.

For additional information that may be helpful, see "Loading a CMS Nucleus" on page 176.

Chapter 6. Installation of Optional Feature Program Products

This chapter provides procedures for all new VM/SP users for installing optional feature program products.

The chapter contains:

1. A procedure for formatting and allocating volumes that are not formatted by the PREP EXEC.

2. An overview of program product installation.

3. Information for using the PPPREP EXEC to format the program product-related minidisks.

4. Information for using the INSTFPP EXEC to install the feature program products from the Feature tape(s) and verify that they are installed correctly.

Notes:

1. The optional feature program product execs (PPPREP and INSTFPP) assume that certain disks exist with certain passwords. You should not delete or alter passwords of minidisks related to the optional feature program products you will be installing.

   After the installation process is complete, you will want to customize your directory by entering unique userid passwords and read/write passwords to the minidisks. This will help provide for data security for your site.

2. Appendix I, "Optional Feature Program Products" on page 287 contains additional information about the optional feature program products.

3. The sample file listings are in:

   Appendix J, "3310 Based System Sample Files" on page 303
   Appendix K, "3330 Based System Sample File" on page 333
   Appendix L, "3340 Based System Sample Files" on page 363
   Appendix M, "3350 Based System Sample Files" on page 395
   Appendix N, "3370 Based System Sample File" on page 425
   Appendix O, "3375 Based System Sample File" on page 455
   Appendix P, "3380 Based System Sample File" on page 487

Formatting and Allocating Volumes (not done by the PREP EXEC)

The PREP EXEC (using the CPFMT parameter) does not format and allocate DASD volumes such as VMPK02, VMPK03, PROFPK, SQLPK, and SQLPK2. This can be accomplished by loading the "IPL FMT" file to your card reader and then IPLing the card reader to start the FORMAT/ALLOCATE program. This should be done from the MAINT userid. Use the following procedures for each of the volumes to be formatted.

vary on cuu

cuu is the real address of the DASD volume.

```
CUU VARIED ONLINE
R;
```

att cuu *

cuu is the real address of the DASD volume.

```
DASD CUU ATTACH TO MAINT CUU
R;
```

pur rdr all

```
nn FILES PURGED
R;
```

sp pun *

pun ipl fmt s (noh

```
PUN FILE NNNN TO MAINT COPY 001 NOHOLD
R;
```

ipl 00c

```
VM/370 FORMAT/ALLOCATE PROGRAM - VM/SP
ENTER FORMAT OR ALLOCATE:
```

format

```
FORMAT FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real address of the DASD volume.

```
ENTER DEVICE TYPE:
```

dt

dt is the device type of the DASD to be formatted. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512. (Use FB-512 for a 3310 or a 3370 system.)

```
ENTER START CYLINDER (NNN) OR "LABEL":
```

When using FBA devices (3310 or 3370), the system will display **ENTER START PAGE (NNN) OR "LABEL"** .

ENTER

```
ENTER END CYLINDER (NNN):
```

When using FBA devices (3310 or 3370), the system will display **ENTER END PAGE (NNN).**

ENTER

```
ENTER DEVICE LABEL:
```

xxxxxx

xxxxxx is your label name, for example VMPK02.

```
FORMAT STARTED
FORMAT DONE
000 NO. PAGE RECORDS WITH READ-CHECK ERRORS
```

If the previous message indicates other than 000 read-check errors, discontinue the installation process and contact your Systems Engineer or hardware service personnel. (You may have to initialize the volume with the Device Support Facility. Refer to the *Device Support Facility User's Guide and Reference.*)

```
ENTER FORMAT OR ALLOCATE:
```

allocate

```
ALLOCATE FUNCTION SELECTED
ENTER DEVICE ADDRESS (CUU):
```

cuu

cuu is the real address of the DASD volume.

```
ENTER DEVICE TYPE:
```

dt

dt is the device type of the DASD to be formatted. Valid entries are: 3330, 3330-11, 3340-70, 3350, 3375, 3380, or FB-512. (Use FB-512 for a 3310 or a 3370 system.)

```
ENTER DEVICE LABEL:
```

xxxxxx

xxxxxx is your label name, for example VMPK02.

```
ENTER ALLOCATION DATA FOR VOLUME XXXXXX
TYPE CYL CYL
```

For FB-512 type DASD, the message will be **TYPE PAGE PAGE.**

Find the correct start and end page/cylinder values for your DASD type in the table below.

CKD	**Cylinder**		**FB-512**	**Page**	
	Start	**End**		**Start**	**End**
3330	000	403	3310	00002	15751
3330-11	000	807	3370-1	00002	69749
3340-35	000	347	3370-2	00002	89093
3340-70	000	695			
3350	000	554			
3375	000	958			
3380	000	884			

Then, enter them in the command that follows:

perm sss eee

sss is the start cylinder or page for DASD type.

end

eee is the end cylinder or page for DASD type.

```
ALLOCATION RESULTS
PERM   XXXXX  XXXXX
DEVICE CUU VOLUME XXXXXX  ALLOCATION ENDED
ENTER FORMAT OR ALLOCATE:
```

Do not respond to this message.

If you are running *first level*:

ATTN For example, press PA1 on 3278 terminal.

to enter the CP environment.

If you are running *second level*, enter

#cp ipl cms

where # represents the linend character of your *second level* system.

Warning: When running a *second level* machine, the *first level* linend character # has to be changed to some other character. Enter TERM LINEND % if you did not do so previously.

detach cuu

cuu is the real address of the DASD volume.

```
DASD CUU DETACHED  MAINT CUU
```

attach cuu system xxxxxx

cuu is the real address of the DASD volume.
xxxxxx is your label name, for example VMPK02.

```
DASD CUU ATTACH TO SYSTEM XXXXXX
```

Program Product Installation Overview

Two execs are provided to assist with installation of program products from the Feature tape(s). The execs are the PPPREP EXEC and the INSTFPP EXEC. The PPPREP EXEC is used to format the program product-related minidisks defined in the sample directory. The INSTFPP EXEC is used to install the desired program products.

Appendix I, "Optional Feature Program Products" on page 287 contains a list of the program products that are included on the Feature tape(s). Other product installation related information is also included in the appendix. Refer to the appendix now and as needed during the installation of the program products.

Using the PPPREP EXEC for Formatting the Program Product-Related Minidisks

Program product-related minidisks must be formatted before program products can be installed. The PPPREP EXEC is provided to assist in the formatting of the minidisks.

The user who will be formatting the minidisks must be logged on as MAINT. All users should be logged off the system while the minidisks are being formatted because MAINT cannot link to a minidisk in write mode if another user is linked to the minidisk in R/W mode.

Be aware that the PPPREP EXEC will format several minidisks on the system. Any data already on those disks will be destroyed. The minidisks that will be formatted are listed below. The system also prints the list for you during execution of the PPPREP EXEC.

Follow the procedures below to format the minidisks.

1. Logon as MAINT if you are not already logged on with the MAINT userid.

   If you have just completed formatting and allocating any volumes (not done by the PREP EXEC), log off the MAINT userid and log on again. This is to ensure that the minidisk links in the MAINT directory entry will be executed correctly.

2. Force all users, including the operator, except MAINT off the system. This must be done because the PPPREP EXEC links to user minidisks in R/W mode. If another user is linked to a minidisk in R/W mode, MAINT will not be able to make the link in write mode.

   Enter the following commands to notify all users that they should log off the system so that service work can proceed.

```
wng all all users will be forced off the system, log off now
```

Tells any user on the system to log off immediately or be forced off.

`query names`	Displays the names of all users on the system.
`force xxxxxxxx`	xxxxxxxx is the name of a system user to be forced off.

This page intentionally left blank.

You must execute the FORCE command for each user on the system, including the operator.

3. Have a copy of the directory handy. If a link fails, you will be asked for the write password of the specific minidisk.

   Note: If your directory has been changed, you should execute DISKMAP EXEC, if you haven't already done so. Output from this exec, displays any minidisk overlaps. These overlaps could cause serious problems during PPPREP.

4. Enter the following command to start the formatting process:

```
  ppprep

Warning:  The PPPREP EXEC will format several minidisks on your system.
Any data already on those disks will be destroyed.  Only use PPPREP
prior to the initial installation of optional feature program products.

THE MINIDISKS WHICH WILL BE FORMATTED ARE:

ADMIN    191                IPCS     191          OPERATOR 191
AUTOLOG1 191                ISMAINT  191          OPERATNS 191/193
CMSBATCH 195                ISPVM    191/192      OP1      191
CMSUSER  191                MAINT    19E          PVM      191
CPRM     191/192/291        MAINT    300          RSCS     191
DATAMOVR 191                MAINT    310          SYSDUMP1 191
DIRMAINT 191/193/195        MAINT    319          VMAP     191
DISKACNT 191                MAINT    31A          VMUTIL   191
EREP     191                MAINT    325
IIPS     191/193            MAINT    3A0

DO YOU WISH TO RUN PPPREP AT THIS TIME? (PLEASE ANSWER YES OR NO)

  yes
```

```
NOTE:  THIS EXEC WILL TAKE SOME TIME TO EXECUTE.

FORMATTING MAINT 19E MINIDISK
FORMATTING MAINT 300 MINIDISK
FORMATTING MAINT 310 MINIDISK
FORMATTING MAINT 319 MINIDISK
FORMATTING MAINT 31A MINIDISK
FORMATTING MAINT 325 MINIDISK
FORMATTING MAINT 3A0 MINIDISK
FORMATTING DIRMAINT 191 MINIDISK
FORMATTING OPERATNS 193 MINIDISK
FORMATTING IIPS 191 MINIDISK
FORMATTING IIPS 193 MINIDISK
FORMATTING VMAP 191 MINIDISK
FORMATTING ISMAINT 191 MINIDISK
FORMATTING PVM 191 MINIDISK
FORMATTING ISPVM 192 MINIDISK
FORMATTING CPRM 191 MINIDISK
FORMATTING CPRM 291 MINIDISK
FORMATTING ADMIN 191 MINIDISK
FORMATTING AUTOLOG1 191 MINIDISK
FORMATTING CMSBATCH 195 MINIDISK
FORMATTING CMSUSER 191 MINIDISK
FORMATTING CPRM 192 MINIDISK
FORMATTING DATAMOVR 191 MINIDISK
FORMATTING DIRMAINT 193 MINIDISK
FORMATTING DIRMAINT 195 MINIDISK
FORMATTING DISKACNT 191 MINIDISK
FORMATTING EREP 191 MINIDISK
FORMATTING IPCS 191 MINIDISK
FORMATTING ISPVM 191 MINIDISK
FORMATTING OPERATOR 191 MINIDISK
FORMATTING OPERATNS 191 MINIDISK
FORMATTING OP1 191 MINIDISK
FORMATTING SYSDUMP1 191 MINIDISK
FORMATTING VMUTIL 191 MINIDISK
COPYING SAMPLE PROFILE EXEC TO CMSUSER 191 MINIDISK
FORMATTING RSCS 191 MINIDISK
CON FILE XXXX TO MAINT COPY 001 NOHOLD
```

Note: An error message will be displayed if any of the above minidisks did not format correctly. To start, examine your directory. Determine why the minidisk did not format properly and correct the problem. You will then have to format the minidisk manually because if you invoke the PPPREP EXEC again, it will format all the minidisks again. (If you chose to delete the definition for a minidisk which PPPREP tries to format, you will receive a message; you may ignore it.)

Using the INSTFPP EXEC for Installing the Feature Program Products

Note: Before using the INSTFPP EXEC to install any optional feature program products, be sure to read "Considerations for Installing Optional Feature Program Products" on page 133.

The INSTFPP EXEC manages the installation of feature program products by doing the following:

- Loads from tape and executes the product installation EXEC.
- Updates the PROD LEVEL file on the MAINT 319 minidisk.
- Prints the product MEMO file and copies it to the MAINT 319 minidisk.

To use the INSTFPP EXEC, you must:

- Log on as MAINT
- Define your storage size as 12M
- IPL CMSL (not CMS).

The INSTFPP EXEC has a help facility that can be invoked by entering HELP INSTFPP.

Have a copy of your directory available. Many of the program product installation execs perform links to the user minidisks in write mode. If the attempt to make the link fails, you may be asked for the write password of the minidisk.

If your printer handles only uppercase characters, you must use the FOLD option of the LOADBUF command. In addition, many of the product memos contain special characters that your printer may not handle. If your printer will not print the special characters, you will have to look at the product memo files online. The product memo files are on the MAINT 319 minidisk.

This page intentionally left blank.

A console file is created during the installation process to record the terminal activity. This file will be spooled to the MAINT reader at the completion of the INSTFPP EXEC.

The feature program product tape is assumed to be at virtual address 181. To check, enter Q V 181. If the tape is not accessed as 181, access it as 181.

> **Note:** If INSTFPP is invoked with no parameters from a 3270 device (20 line minimum), a set of user assistance panels is displayed.
>
> There are HELP screens available to explain the INSTFPP panels. These screens are invoked when the user presses PF1 from any panel.

INSTFPP command syntax is as follows:

INSTFPP	[NOPROMPT] [MEMO] [ALL I prodid(prodid...)]

NOPROMPT

eliminates the prompt that requires the user to specify whether or not to install the specified program product.

MEMO

allows printing of the product MEMO files from the tape **without** installing the program products. If MEMO is not used, the program products are installed. The product MEMO file is always printed when a program product is installed. You can find the instructions for manual verification and product installation in the product MEMO files.

ALL

allows the user to process all the program product(s) on the tape. If INSTFPP ALL is entered, all the program products on the tape will be installed, each product MEMO file is printed and the user is prompted before each program product is installed.

prodid(prodid...)

allows the user to specify one or more program product IDs when INSTFPP is invoked. INSTFPP then scans the EXECs on the stacked tape and installs the selected program products.

Note: To restart program product installation:

- Rewind and ready the tape.
- Issue **INSTFPP**.
- Select only those products that did not get installed successfully.

After all program products are installed, follow the instructions in each product's MEMO file to verify that they were installed correctly. You do not have to do this for products that were automatically verified during installation.

PROD LEVEL File

The INSTFPP EXEC updates a file named "PROD LEVEL" on the MAINT 319 minidisk to indicate the results of each program product installation.

The following is an example of what a PROD LEVEL file may look like:

```
* 5748XXB - DISPLAY MANAGEMENT SYSTEM (DMS/CMS) *
RELEASE 2.0
TIME AND DATE OF ENTRY: 14:22:12 83/10/18
*** PRODUCT INSTALLED AND VERIFIED SUCCESSFULLY

* 5798DMY VM FILE STORAGE FACILITY *
VERSION 1  RELEASE 1  MODIFICATION 0
TIME AND DATE OF ENTRY: 19:16:27 83/10/19
*** PRODUCT INSTALLED; MUST BE MANUALLY VERIFIED
```

Update Messages

Each program product entry in the PROD LEVEL file has an update message associated with it. The possible update messages and their explanations are:

***** PRODUCT INSTALLED AND VERIFIED SUCCESSFULLY**

The program product installed correctly and the product was verified successfully.

***** PRODUCT FILES LOADED; SEE PRODUCT MEMO TO COMPLETE INSTALLATION**

The program product files have been loaded successfully. You must refer to the product memo that was printed out by the INSTFPP EXEC. This memo will tell you how to complete the installation of the product and then verify that it installed correctly. In some cases the product memo may refer to other documentation.

***** PRODUCT INSTALLED; MUST BE MANUALLY VERIFIED**

The program product installed but it was not verified automatically. You must refer to the product memo that was printed out by the INSTFPP EXEC. This memo will tell you how to verify that the program product was installed correctly. In some cases the product memo may refer to other documentation.

***** PRODUCT INSTALLED; VERIFICATION FAILED**

The program product installed but the automatic verification failed. You can try to install the program product again and if it still does not verify correctly, contact your support personnel.

***** PRODUCT INSTALLATION EXEC FAILED, RC = XXX**

The program product installation exec failed. The return code passed back by the program install exec to the INSTFPP EXEC is indicated. Look at the program product installation exec to see what the specific return code means. If you cannot fix the problem, contact your support personnel.

Considerations for Installing Optional Feature Program Products

Some special considerations must be taken into account before several of the program products are installed. These are:

- Many program products will not install properly without CMS Large (CMSL). IPL CMSL before installing optional program products.

- MAINT's 319 (P) and 325 (W) minidisks are NOT able to hold ALL the possible optional program products. You may want to create a separate disk for them. Refer to Figure 26 on page 290 for the space needed for each program product.

- All program products will verify from the CMSUSER userid except for the ones noted here. Refer to the individual product memos for specific instructions on what userid(s) are used for verification. These program products are:

  - CSP/AD (5668-944)
  - CSP/AE (5668-945)
  - DIRMAINT (5748-XE4)
  - EP/VS (5744-AN1)
  - INFO/SYS (5735-OZS)
  - IPF (5748-MS1)
  - PVM (5748-RC1)
  - RSCS (5748-XP1)
  - SMART (5796-PNA)
  - SQL/DS (5748-XXJ)
  - VMAP (5798-CPX)
  - VMBATCH (5796-BCY)

- As indicated in "Considerations for Installing Optional Feature Program Products," both Information System (INFO/SYS) and VM/Interactive File Sharing (VM/IFS) require VSE/VSAM. Because the three products are interrelated, you must manually verify, in order, the installation of each. The precise order to follow is documented in the related *Memo to Users*.

  Also, these products depend on a temporarily formatted minidisk called Master Catalog. Release this disk upon completion to avoid problems with other products that may use the system temporary workspace.

- During the install process for IPF, no messages confirming the loading or verifying of the product files are displayed. Therefore, to be sure all files have loaded successfully, examine the PROD LEVEL file located on MAINT's 319 minidisk.

- ADRS-II (5796-PLN) requires VSAPL (5748-AP1) to be installed first.

- ADRS-II (5796-PLN) with 'Full Screen Support' requires GDDM (5748-XXH) to be installed before VSAPL (5748-AP1).

- ADRS-II/BG (5796-PLN) with 'Business Graphics' requires GDDM (5748-XXH) to be installed before VSAPL (5748-AP1).

- APL/DI-II (5796-PNG) requires VSAPL (5748-AP1) and PL/I Transient Library (5734-LM4) to run.

- CSP/AD (5668-944) requires CSP/AE (5668-945) and VSE/VSAM (5746-AM2) to be installed first.

- CSP/AE (5668-945) requires VSE/VSAM (5746-AM2) to be installed first.

- EREP (5749-010) when you verify the installation of this product using the instructions found in the EREP MEMO file, be sure to access the disk containing the V5749010 EXEC as filemode A.

- FPS-II (5798-DCN) requires VSAPL (5748-AP1).

- INFORMATION/SYSTEM (5735-OZS) requires VSE/VSAM (5746-AM2). Installation of the INFORMATION/SYSTEMS-VSE Data Feature requires additional space as specified in the INFORMATION/SYSTEMS Memo to users.

- IPCS/E requires that the CPRM disconnected virtual machine must be logged off before installation of IPCS/E. Users should be informed of this. After the installation and verification of IPCS/E and the creation of IPCSMAPS, the CPRM disconnected virtual machine may be autologged.

  The CP and CMS nucleus maps must be located on the default MAINT minidisks prior to installing IPCS/E. If the nucleus maps were not saved during the base installation, you must save them now. To generate and save the nucleus maps on the default MAINT minidisks, enter the following commands:

  ```
  GENERATE { CP  } NUCLEUS
           { CMS }

  GENERATE MAP
  ```

  It may be necessary to allocate a larger OPERATNS 193 minidisk for dump processing depending upon your Virtual Machine storage size. To calculate the needed space, refer to the *IPCS/E General Information*, GC34-2019 manual.

- ISPF/PDF (5764-172) requires ISPF (5668-960) to be installed first.

- PROFS (5664-176) will need the volume PROFPK formatted and allocated before the product can be installed. This product also requires DCF (5748-XX9) to be installed first. You should print and review the PROFS memo file before installing PROFS. Refer to "Formatting and Allocating Volumes (not done by the PREP EXEC)" on page 125 for information about formatting and allocating volumes. Refer to "Using the INSTFPP EXEC for Installing the Feature Program Products" on page 130 for information about printing product memos.

- SQL/DS (5748-XXJ) If you are installing SQL/DS and are not using VM/SP sample directory entries, you should refer to the *SQL/Data System Installation - VM/SP*, SH24-5044 manual. That manual describes the virtual machines and minidisks that you must define before loading the SQL/DS distribution tape files to disk.

If you are using VM/SP sample directory entries, you must format and allocate the SQLPK volume before installing SQL/DS. In addition, if you are using a 3310 or 3340 DASD configuration, you must also format and allocate the SQLPK2 volume before installing SQL/DS. Refer to "Formatting and Allocating Volumes (not done by the PREP EXEC)" on page 125 for information about formatting and allocating volumes.

When you use the INSTFPP EXEC to load the SQL/DS files from feature tape to disk, the SQL/DS installation exec (I5748XXJ EXEC) is executed. The I5748XXJ EXEC requires write access to the SQL/DS Production Minidisk (SQLDBA 195) and SQL/DS Service Minidisk (SQLDBA 193). To successfully load the SQL/DS files from the feature tape to disk, the MAINT userid cannot already have a 193 or a 195 disk defined. Also, if MAINT has any disks accessed as filemodes V or Q, I5748XXJ EXEC releases them when it begins running and reaccesses them before exiting.

This page intentionally left blank.

Chapter 7. Introduction to VM/SP Service

Updating Your VM/SP System

When you have a servicing task, you want to do it as quickly as possible without excessive delay or unnecessary steps. VM/SP provides you with several procedures and techniques for updating your VM/SP system.

In most cases, servicing of your VM/SP system is done by applying a system Program Update Tape (PUT). The PUT is distributed regularly to each IBM customer. Application is an automatic process carried out by following the detailed instructions included in the PUT Document which comes with every PUT.

There are instances, however, when you need to update an individual module or apply a fix that is not included on the PUT. VM/SP provides you with EXEC procedures and programs to aid you in accomplishing this task. Using a virtual machine, you can do updating and servicing tasks concurrently with other production work.

The framework provided by VM/SP gives you a maximum amount of flexibility in servicing your system. This framework includes:

- A system support plan
- A system Program Update Tape
- Naming conventions for update files and control files
- Several EXEC procedures, commands, and programs. (Figure 16 on page 151 lists the programs.)

Procedures are provided, under CMS, to apply updates to modules and create new load (TEXT) files, to be included in a new CP or CMS nucleus. These procedures maintain the integrity of original source files and previously applied modifications. The update structure involves file naming conventions for update and text files, a set of programs to support the processing, and several EXEC procedures to process the files.

Any non-PUT updates must also be applied to a copy of the original source file. At the same time, any previously applied updates must be repeated. The source can then be assembled and the new text file included in either the CP or CMS system.

All these techniques require the use of CMS. You should have a thorough understanding of the CMS file system and disk search order, the EXEC interpreters, and the UPDATE command before you try to use any of the procedures described.

The *VM/SP CMS User's Guide* provides information on CMS. For reference material on CMS commands and EXEC control statements, refer to the *VM/SP CMS Command and Macro Reference.*

System Program Update Tape (PUT)

IBM regularly distributes a system Program Update Tape (PUT) containing service updates for the VM/370 system control program, VM/SP, and other program products for which you are licensed. Service for VM/SP is cumulative and complete, containing all current and prior service. When applying service to VM/SP, you must *not* apply service to the VM/370 Release 6 IPCS and RSCS components as neither of these products is a discrete component of VM/SP.

VM/System Product service is initiated by the VMSERV EXEC. VMSERV is always supplied in the first tape file of the system PUT. The VMSERV EXEC controls the individual service EXECs that apply service to build a new CP or CMS nucleus from the replacement text decks on the system PUT.

The first file of the system PUT also contains the PUT Document, which describes the PUT and details for the service application process. VMSERV prints the PUT Document and the *Memo to Users*.

The second file on the system PUT contains the *Memo to Users* file(s) for each product that contains service on the PUT volume. This file is loaded on your C-disk and printed by VMSERV. The *Memo to Users* provides step-by-step instructions for applying the updates. It also tells you when you must do additional steps before invoking VMSERV to build a new nucleus. For example, if a macro library has been updated and your system definition files (DMKRIO, DMKSYS, and so on) must be reassembled, the user memo tells you to use the VMFASM EXEC to reassemble the source files.

A Virtual Machine for Updating VM/SP

The VM/SP directory distributed with each Starter System contains an entry for a userid MAINT. You may want to use this userid for system updating and servicing. MAINT's virtual machine should have access to all the disks required for system service.

If you foresee the need to apply your own local updates to VM/SP, you need to add an additional MDISK statement to MAINT's directory entry. The procedure in this section assumes a user defined minidisk address of 295 for local service.

The entries in the VM/SP directory, with the exception of the 293, 294, 393, and 394 virtual disks, are included in the 3330, 3340, 3350, 3375, 3380, and FB-512 VM/SP directories supplied with the Starter System. They should be included in your VM/SP directory, as they are used by IBM for support.

The contents of the following virtual disks are:

Disk	Contents
190	Current CMS system disk
191	Work area
194	CP text retention
19D	HELP files
201	EREP txtlib
293	CMS PTFs, updates, and updated text decks (object modules). The updated text decks may not be present in some installations.
294	CP PTFs, updates, and updated text decks (object modules)
295	User defined minidisk for locally applied service.

393	CMS source and macros
394	CP macros, copy files, and source
cuu	CP system residence device, or a replica of it, for test purposes

These virtual disks are shown in Figure 14.

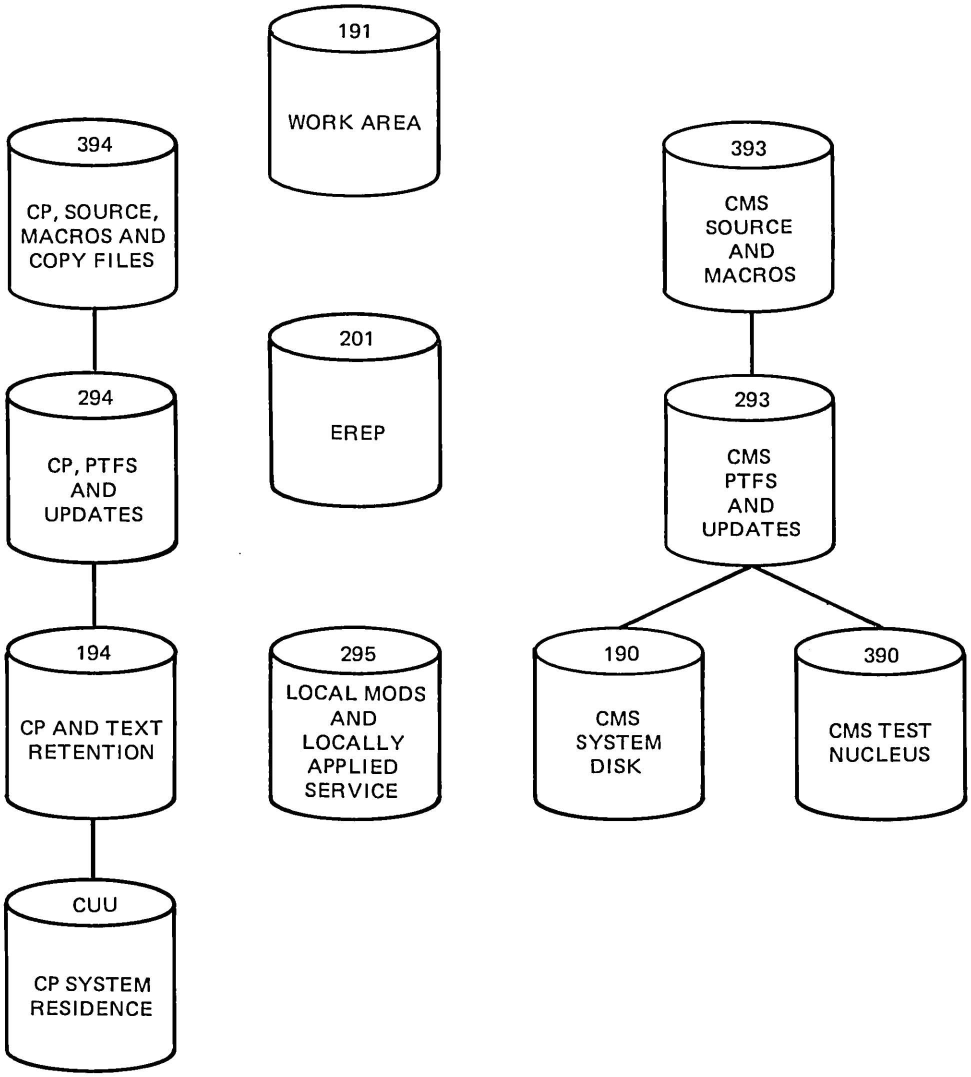


Figure 14. System Support Plan

Suggested VM Configuration for Updating

The following is a sample of a suggested virtual machine configuration for updating. Use the sample for your device type as a guide when applying service.

Example for Updating a 3330 System

```
USER MAINT CPCMS 6M 16M ABCDEFG
 ACCOUNT 1 SYSPROG
 IPL 190
 CONSOLE 009 3215
 SPOOL 00C 2540 READER *
 SPOOL 00D 2540 PUNCH  A
 SPOOL 00E 1403 A
 MDISK 123 3330 000 404 VMSRES MW RSYSRES WSYSRES MSYSRES
 MDISK 124 3330 000 404 VMSEXT MW RSYSRES WSYSRES MSYSRES
 MDISK 125 3330 000 404 VMPK01 MW RSYSRES WSYSRES MSYSRES
 MDISK 126 3330 000 404 VMSTGE MW RSYSRES WSYSRES MSYSRES
 MDISK 190 3330 098 100 VMSRES MW ALL     WMAINT  MMAINT
 MDISK 191 3330 021 012 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 194 3330 033 045 VMSRES MW RMAINT  WMAINT  MMAINT
 MDISK 295 3330 xxx nnn VMSRES MW RMAINT  WMAINT  MMAINT[14]
 MDISK 201 3330 021 045 VMPK01 MW RMAINT  WMAINT  MMAINT
 MDISK 293 3330 335 024 VMPK01 MW RCMSAUX WCMSAUX MCMSAUX
 MDISK 294 3330 359 023 VMPK01 MW RCPAUX  WCPAUX  MCPAUX
 MDISK 393 3330 001 130 VMSTGE WR RMAINT  WMAINT
 MDISK 394 3330 234 170 VMSTGE WR RMAINT  WMAINT
 MDISK 19E 3330 195 090 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 319 3330 135 060 VMSEXT MW ALL     WMAINT  MMAINT
 MDISK 19D 3330 331 060 VMSRES MW ALL     WMAINT  MMAINT
 MDISK cuu 3330 000 404 yyyyyy MW[15]
```

Figure 15. Suggested VM Configuration for Updating a 3330 System

Files for System Updates

Each of the components of VM/SP has a unique character module identifier, which is used to name the component's modules. These module identifiers are also used to name the files that are used to update the components. The identifiers are:

Component	Module Identifier
CP	DMK
CMS	DMS

The default CMS filetypes are used to identify the source, object code, module files and libraries associated with each component. These filetypes are:

Filetypes	Type of File
ASSEMBLE	Source File
TEXT	Object deck (relocatable)
TXTAP	Object deck with attached processor support (relocatable)
TXTMP	Object deck with multiprocessor support (relocatable)
MODULE	Nonrelocatable object code
MACLIB	Macro or copy library
TXTLIB	Text file library

[14] xxx is the starting address and nnn is the size of the minidisk allocated for your local mods and locally applied service.

[15] cuu and yyyyyy are the address and label of your system residence volume defined in your DMKSYS module.

Chapter 9. Updating An Installed VM/SP System

Building a New CP Nucleus

If you are going to use the MAINT userid to load and test a new CP nucleus, you should be sure that MAINT's virtual machine has:

- A minimum 1M of virtual storage. The loader requires 512K to execute. In general, MAINT's virtual machine should have as much virtual storage as the real machine storage size.

- The ECMODE option specified in the VM/SP directory (or has used the CP SET ECMODE ON command). ECMODE is required for testing the CP system in your virtual machine.

- Write access to the CP system residence volume (SYSRES), or a minidisk that is a replica of SYSRES.

  Notes:

  1. The minidisk must be defined in your virtual machine at the same address as the real address of the SYSRES.

  2. The minidisk must have been formatted with the CP Format/Allocate program, so it resembles the CP system residence.

  3. Unused cylinders beyond the extent of the minidisk must be allocated as permanent space (PERM).

To build a CP nucleus, spool the punch continuous to yourself and invoke VMFLOAD to punch to your reader a file containing the CP loader followed by all the textfiles for the nucleus. Then, IPL the loader file. IPL gives control to the loader. The loader reads in the rest of the deck, link-edits the textfiles, and prints the load map. It then passes control to DMKSAVNC, which writes the nucleus out onto the SYSRES volume described in DMKSYS.

If you followed the steps in the preceding chapter under "Using VMFLOAD To Punch a New Nucleus," the stand-alone loader (DMKLD00E LOADER) is in your card reader. It is followed by all the text decks necessary to build a CP nucleus. There are several ways to handle this reader file.

- Perhaps the most common delivery method is to move the reader file to a tape. Later, IPL the loader from that tape into the real machine and write the nucleus to the real SYSRES.

  To move the reader file to tape, use the following commands:

  ```
  tape rew
  filedef input reader
  filedef output tap1 ( recfm f lrecl 80 block 80
  movefile input output
  ```

- You can get write access to the real SYSRES and IPL the reader file in your virtual machine to write the nucleus out onto your real SYSRES. This is risky because you are writing over the existing nucleus on your SYSRES. If there are errors in your new nucleus, the next time the real system is shutdown and reIPLed you may encounter disastrous and unpredictable results.

- You can set up a virtual SYSRES on a minidisk, giving it the same address and volid as the real SYSRES and CP allocating it. Then, IPL the reader file in your virtual machine to cause the loader to write the new nucleus onto the virtual SYSRES. After you have tested the new nucleus in your virtual machine and are happy with it, you can use DDR COPY NUC command to copy it from your virtual SYSRES to the real one.

  The DDR COPY NUC command might also be used to move the nucleus from the virtual SYSRES to tape. After which you shut the real system down at some point and install the new nucleus using the RESTORE NUC function of stand-alone DDR.

Example of New CP Nucleus Build

For this example, the third method described above is used to handle the reader file. It is assumed that MAINT's virtual machine has entries for the real system residence volume at address 123 and for a minidisk replica at address 331. You need to detach the real SYSRES and define the minidisk at that address by entering:

```
detach 123

define 331 as 123
```

When you IPL the nucleus, the load map is spooled to your virtual printer. To retain a copy of the load map as a CMS disk file, issue the command:

```
spool printer to *
```

The load map is routed to your card reader. The CMS READCARD command can be used later to write the load map on disk. (Remember to close the spool file by issuing CLOSE 00E after the IPL has completed.)

Now you can IPL the CP nucleus, specifying the address of your virtual card reader.

```
ipl 00c
```

```
NUCLEUS LOADED ON SYSRES- STARTING CYL/BLK = 397,
LAST CYL/BLK USED = 401
```

The above message gives the starting and ending cylinders/blocks for the nucleus. Verify that this nucleus has not expanded into another allocated area such as "TEMP," "WARM START," etc., or unpredictable results may occur.

The message below does not automatically display on many processors. Check your processor reference manual for information about displaying and interpreting the PSW. If the PSW message is not displayed, press PA1 to get a CP READ state, then proceed.

```
CP ENTERED; DISABLED WAIT
PSW '00020000 00000012'
```

This is the normal return if the nucleus loaded correctly. If the PSW displayed is not "00020000 00000012," refer to the CP WAIT state codes in *VM/SP System Messages and Codes.*

Now define your console address to be the same as defined in the RIOGEN macro in DMKRIO.

```
   def 009 as cuu
CONS CUU DEFINED
```

cuu is the console address in RIOGEN macro in DMKRIO.

Then, IPL the replica system residence device by issuing:

```
   ipl 123
VM/SP RELEASE r SERVICE LEVEL nn; mm/dd/yy hh:mm:ss

NOW hh:mm:ss EST weekday mm/dd/yy
CHANGE TOD CLOCK (YES|NO)

   no
hh:mm:ss START ((COLD|WARM|CKPT|FORCE) (DRAIN|SHUTDOWN))

   shutdown
DMKCPI960W SYSTEM WARM START DATA SAVED

DMKCPI961W SYSTEM SHUTDOWN COMPLETE

DMKDSP450W CP ENTERED; DISABLED WAIT PSW
CP
```

After you check the new CP, you may redefine your console by issuing:

```
   def cuu as 009
CONS 009 DEFINED
```

CMS accepts only 009 and 01F as valid console addresses.

IPL the CMS system and issue the DDR command to create a backup copy of the CP nucleus.

```
   ipl cms
CMS...mm/dd/yy
  ddr
ENTER:
   in 123 3380 sysres
```

The input unit is defined using the address of your replica system residence device.

```
ENTER:
   out 181 3420
```

Your output unit is the tape drive that you have attached to your virtual machine at address 181.

When you enter the DUMP statement with the NUCLEUS operand, DDR creates a copy of the nucleus that was just loaded.

```
ENTER:
   dump nuc
DMKDDR711R VOLID READ IS 'volser'
DO YOU WISH TO CONTINUE? RESPOND YES, NO OR REREAD:
   yes
ENTER NEXT EXTENT OR NULL LINE
ENTER:
   ENTER
DUMPING 'volser'
END OF DUMP
ENTER:
   ENTER
END OF JOB
PRT FILE nnnn FOR userid COPY 001 NOHOLD
R;
```

You have created a backup copy of the CP nucleus. This copy may later be restored using the stand-alone version of the DDR program on the real machine.

Generating and Saving the CMS Nucleus Large Saved Segment

Assuming you are logged on using the MAINT userid, enter:

define storage 16m

ipl 190

generate cmsl nucleus

The system responds with the following:

```
VMFLOAD BEING EXECUTED
LOADLIST = CMSLOADL AND CNTRL = DMSSP
SYSTEM LOAD DECK COMPLETE
PUN FILE nnnn TO MAINT COPY 001 NOHOLD

THE LOADMAP FOR THE CMS NUCLEUS SHOULD BE SAVED ON MDISK 191,
WHEN THIS STEP COMPLETES, RE-IPL THE CMS SYSTEM (I 190)
AND ENTER: GENERATE MAP

DMSINII606R SYSTEM DISK ADDRESS =
```

190

```
DMSINI615R Y-DISK ADDRESS =
```

19e

```
DMSINI640R HELP DISK ADDRESS =
```

19d

```
DMSINI607R REWRITE THE NUCLEUS ?
```

yes

```
DMSINI608R IPL DEVICE ADDRESS =
```

190

The response to this prompt is the address of the device where the CMS nucleus is to be written.

```
DMSINI609R NUCLEUS CYL/BLK ADDRESS =
```

97

The response to this prompt is device dependent. See the table below for the correct CMSL value for your device type.

Device	CMS	CMSL
FB-512	43520 blk	44544 blk
3330	094 cyl	097 cyl
3340	222 cyl	229 cyl
3350	045 cyl	047 cyl
3375	070 cyl	072 cyl
3380	043 cyl	044 cyl

```
DMSINI610R ALSO IPL CYL/BLK 0 ?
```

no

```
DMSINI611R VERSION IDENTIFICATION =
```

ENTER

You are permitted to enter up to 32 descriptive characters to identify the version and level of CMS. This information is displayed each time you IPL the CMS system now being generated.

```
DMSINI612R INSTALLATION HEADING =
```

You are permitted to enter up to 64 descriptive characters to serve as an installation standard heading at the beginning of each output file.

ENTER

```
SYSTEM SAVED
VM/SP REL n  mm/dd/yy hh:mm
```

ENTER

```
R;
```

generate map

```
LOAD MAP HAS BEEN PLACED IN READER

SELECT YOUR NUCLEUS MAP PREFIX - ENTER UP TO FIVE LETTERS
EXAMPLE: CP, CMS, CMSL, _____?
MINIDISK DEFAULT IS 194, CMS AND CMSL DEFAULT IS 191.
TO CHANGE MINIDISK DEFAULT USE FORM 'PREFIX CUU' (EX: CMS 194)
NOTE: ENTER NULL RESPONSE OR EXIT TO END
```

cmsl

```
'191' REPLACES ' A (191) '
YOU HAVE NOW SAVED "CMSLNUC MAP" ON MDISK "191"
R;
```

Note: In the procedure that follows, be sure to enter the SAVESYS CMSL command before ENTER. (A null entry usually causes IPL execution to complete.)

To save the CMSL system, enter:

ipl 190 97 clear

97 is the cylinder number you used in response to message number DMSINI609R. The CLEAR option assures that the nucleus has a storage key of zero.

When the *initial* IPL message displays, enter:

savesys cmsl

If the CMS system is named something other than CMS see the *VM/SP System Programmer's Guide.*

To complete the IPL process:

ENTER

```
SYSTEM SAVED
VM/SP REL n  mm/dd/yy hh:mm
```

Your CMSL system is now saved. You can issue IPL CMSL instead of IPL 190, when you wish to run CMSL.

Chapter 10. EXEC Procedures and Command Format Summaries

The command formats, options, and operands for each of the updating EXEC and command procedures are described next, in alphabetical order.

ASMGEND

Use the ASMGEND EXEC procedure to build the system assembler and to create the associated auxiliary directory. ASMGEND loads the text decks for the assembler in the correct overlay structure and produces a load map.

Notes:

1. The assembler text decks normally reside on the system S-disk in file mode S1. This disk must be accessed in some additional file mode prior to issuing this command in order to locate these files. For example:

   ```
   acc 190 a
   ```

2. Use the ASMGEND EXEC if you have modified the assembler (IFOXnn) source. If you have not modified this source, and wish to create the assemble module, possibly after modifying DMSASM or creating a new CMS system disk, use the CMSGEND EXEC.

The format of the ASMGEND command is:

ASMGEND	

Responses

The ASMGEND EXEC procedure displays the following status and error messages:

```
ENTER TARGET DISK MODE FOR ASSEMBLE MODULES
DEFAULTS TO S-DISK IF NONE ENTERED
```

You enter the mode letter of the disk containing the modules referred to from the auxiliary directory. If you enter a mode letter, ASMGEND uses that mode letter as the "targetmode" operand of the GENDIRT command when it creates the auxiliary directory. If you do not specify a mode letter, S is used.

```
ASMGEND XF GEND COMPLETE
```

This message indicates that the system assembler and its associated auxiliary directory are generated successfully.

```
ASMGEND XF GEND FAILED
```

This message indicates that the system assembler text files were not loaded successfully.

CMSGEND

Use the CMSGEND EXEC procedure to generate a new CMS module or LOADLIB from a text file and place the new CMS module or LOADLIB on the specified disk.

Notes:

1. The assembler text decks normally reside on the system S-disk in file mode S1. This disk must be accessed in some additional file mode prior to issuing this command in order to locate these files. For example:

   ```
   acc 190 a
   ```

2. Use the CMSGEND EXEC if you have not modified the assembler (IFOXnn) source, and wish to create the assemble module, possibly after modifying DMSASM or creating a new CMS system disk.

3. You can also use the CMSGEND EXEC to regenerate the ASSEMBLE command when you move the CMS system disk. When you specify ASSEMBLE, CMSGEND prompts you to enter a disk mode letter so it can refresh the assembler's auxiliary directory. (Use the ASMGEND EXEC procedure if you are updating the assembler.)

4. When using CMSGEND EXEC to regenerate the PROP command, you are only generating the PROPLIB LOADLIB. (The CMSGEND options NOCLEAR, MAP, and NOINV have no effect when generating the PROP command.)

The format of the CMSGEND EXEC command is:

```
CMSGEND          fn  [ CTLCMS   ]    [ MODE fm ]
                     [ CTLALL   ]    [  A      ]
                     [ NOCLEAR  ]
                     [ MAP      ]
                     [ NOINV    ]
```

where:

`fn` is the filename of the CMS module or LOADLIB that is to be generated by the CMSGEND EXEC. Only one filename may be specified in the CMSGEND command line.

The filenames that may be specified in the CMSGEND command are any disk-resident CMS commands and service programs.

`CTLCMS` displays each CMS command as it is executed in the CMSGEND EXEC procedure. This is equivalent to the EXEC statement &CONTROL CMS.

`CTALL` displays every executable statement as it is executed in the CMSGEND EXEC procedure. This is equivalent to the EXEC statement &CONTROL ALL.

`NOCLEAR` specifies that the CLEAR option is not to be issued when CMSGEND invokes the LOAD command.

`MAP` specifies that the NOMAP option is not to be issued when CMSGEND invokes the GENMOD command.

`NOINV` issues the NOINV option when CMSGEND invokes the LOAD command; this suppresses the displaying of invalid cards at the terminal. If the text deck was created with the VMFASM EXEC, it may contain update listing information; these records are displayed during the loading process unless you specify NOINV.

`MODE fm` indicates the access mode of the disk to receive the new module. Filemode *A* is the default.

How CMSGEND Works

CMSGEND keeps a list of the CMS disk-resident modules and LOADLIBs, the filenames of the text files used to create them, and any special attributes required to generate them. For example, the RELEASE command must be generated with the ORIGIN TRANS and the SYSTEM options. It is composed of the text files DMSARE and DMSALU. To generate a new RELEASE module, you issue:

```
cmsgend release
```

you may receive messages such as the following:

```
*** CURRENT STATUS:
FILE ' RELEASE MODULE A2' EXISTS
FILE ' RELEASE MODOLD A1' DOES NOT EXIST

*** LOADING:
 INVALID CARD - X9999DMS - (PTF description)
         .
         .
         .
 DMSARE   SD 00E000

*** RESULTS:
FILE ' RELEASE MODULE A2' RENAMED TO ' RELEASE MODOLD A1'
' RELEASE MODULE A2' CREATED FROM TEXT DECK ( S ) DMSARE
WITH OPTIONS TRANS SYSTEM NOMAP
```

Responses

The CMSGEND EXEC procedure displays status and error messages.

```
*** CURRENT STATUS:

['fn MODULE A2' EXISTS        ]
['fn MODULE A2' DOES NOT EXIST]
['fn MODOLD A1' EXISTS        ]
['FN MODOLD A1' DOES NOT EXIST]
```

This message indicates whether a generated module already exists.

```
*** LINK EDITING: fn TEXT
```

This message indicates that CMSGEND is link editing fn TEXT to create a LOADLIB. The existing LOADLIB is erased and not renamed when generating a new one.

```
*** LOADING:
```

This message indicates that CMSGEND is loading the text decks.

```
*** (UNDEF. NAMES NORMAL FOR EDMAIN)
*** NOW WE HAVE A SECOND PASS FOR EDMAIN MODULE.
```

These messages indicate that the EDIT command requires two passes to resolve undefined names.

```
*** NOW WE HAVE A SECOND PASS FOR DMSTPF MODULE.
*** NOW WE HAVE A THIRD PASS FOR DMSTPG MODULE.
```

These messages indicate that the TAPE command requires three passes to generate the necessary modules.

```
*** RESULTS:
['fn MODOLD A1' WAS ERASED                    ]
['fn MODULE A2' RENAMED TO 'fn MODOLD A1'     ]

['fn MODULE A2' CREATED FROM TEXT DECK(S) ...]
[ WITH OPTIONS ...                           ]
```

These messages indicate which existing modules were erased and renamed, which text files were used to create the new module, and the attributes used to create the module.

```
ENTER GENDIRT TARGET DISK MODE LETTER
( NULL LINE DEFAULTS TO 'S' DISK )
```

This message is issued when you specify ASSEMBLE. You should enter the mode letter of the disk that contains the system assembler. This letter is used as the target disk mode address for the GENDIRT command.

```
*** ERROR MESSAGE ISSUED IS NORMAL FOR LINK EDITING
```

If the TEXT deck was created with VMFASM EXEC, it may contain update listing information; these records will cause the linkage editor to generate an error message. The error is normal.

Error Messages

```
ERROR OCCURRED. CMSGEND STOPS.
```

This message indicates that an error occurred and that CMSGEND is terminated.

```
INVALID ARGUMENT fn
```

This message indicates an invalid filename was specified on the command line.

VMFASM

Use the VMFASM EXEC procedure to update a specified source file according to entries in a control file, and to assemble the updated source file. VMFASM invokes the CMS UPDATE command. The format of the VMFASM command is:

```
VMFASM         fn ctlfile [(options...)]

               Options:

               [DISK ] [TERM  ] [LIST  ]
               [PRINT] [NOTERM] [NOLIST]

               [DECK  ] [RENT  ] [EXP] [XREF] [MAX]
               [NODECK] [NORENT]              [MIN]
                                              [STD]
```

where:

fn — is the filename of the source file to be updated. It must have a filetype of ASSEMBLE.

ctlfile — is the filename of the control file. The control file must have a filetype of CNTRL.

Options: VMFASM only accepts the nondefaulted options. All other assembler options entered are ignored and the defaults are used.

DISK — places the listing file on a virtual disk.

PRINT — writes the listing file to the printer.

TERM — writes the diagnostic information on the SYSTERM data set. The diagnostic information consists of the diagnosed statement followed by the error message issued.

NOTERM — suppresses the TERM option.

LIST — produces an assembler listing.

NOLIST — does not produce an assembler listing.

DECK — writes an object module on the device specified on the FILEDEF statement for PUNCH.

NODECK — suppresses the DECK option.

RENT — checks the program for a possible violation of program reenterability. Code that makes the program nonreenterable is identified by an error message.

NORENT — suppresses the RENT option.

EXP expands printing of certain macros which check for the SUP parameter issued via the SYSPARM option of the ASSEMBLE command. The default is SUP.

XREF causes the XREF(FULL) option to be invoked when VMFASM invokes the assembler. The default for VMFASM is XREF(SHORT).

MAX causes the BUFSIZE(MAX) option to be invoked when VMFASM invokes the assembler. If both MAX and MIN are specified, then MIN takes precedence.

MIN causes the BUFSIZE(MIN) option to be invoked when VMFASM invokes the assembler. If both MAX and MIN are specified, then MIN takes precedence.

STD causes the BUFSIZE(STD) option to be invoked when VMFASM invokes the assembler. This is the default for VMFASM when neither MAX nor MIN is selected.

How VMFASM Works

The steps taken by the VMFASM EXEC are summarized below.

1. The VMFASM EXEC calls the UPDATE command with the CTL, STK, PRINT, and OUTMODE A1 options.

   UPDATE uses the control file (ctlfile CNTRL) to update the assembler language source file. If updates exist, the new file is named $fn ASSEMBLE. Otherwise, it is called fn ASSEMBLE.

   UPDATE stacks information from the control file in the console stack, and prints the update log file.

   The option OUTMODE A1 specifies that the files created by the UPDATE command will be written onto the A disk. The filemode will be 1.

   If a PTF file is missing, a message stating this will be written to both the console and to the update log file.

2. Using the library list from the MACS record in the control file, VMFASM issues a GLOBAL MACLIB command.

3. The updated source file, $fn ASSEMBLE, is assembled using the options indicated on the VMFASM command line.

4. The output text deck from the assembly, $fn TEXT, is concatenated with the UPDATES file so that the text deck contains a history of update activity.

5. Using the update level identifier from the control file (the identifier of the most recent update that was found and applied is stacked by the UPDATE command), VMFASM determines how to rename $fn TEXT.

   If the update level identifier is TEXT, the text deck is renamed fn TEXT.

If the update level identifier is anything other than TEXT, the text deck is renamed fn TXTxxxxx (where xxxxx is the 1 to 5 character update level identifier). The new fn TEXT (or TXTxxxxx) is found on the 'A' disk.

Note: The new TEXT deck will be filemode A1 regardless of the filemode of the original TEXT deck. If the filemode of the original TEXT deck is filemode 2, the TEXT deck created by VMFASM will have to be renamed to filemode 2.

6. Temporary files ($fn ASSEMBLE or fn ASSEMBLE, fn UPDATES, and fn CTLFILE are erased.

A summary of the input and output files used by VMFASM follows.

DISK INPUT FILES:

fn ASSEMBLE	Assembler Language source
ctlfile CNTRL	control file

MACLIBS, auxiliary control files (AUXxxxxx), and miscellaneous update files.

DISK OUTPUT FILES:

fn {TEXT TXTxxxxx}	object deck, named according to the update level identifier in the control file

This file also contains data from the UPDATES file, together with date and time information.

PRINTER OUTPUT FILES

fn ctlfile	Assembler listing (if PRINT options is in effect)

This file also contains data from the update log file (fn UPDLOG), describing the updates applied to the source file.

Responses

The UPDATE command issues the message DMSUPD178I to indicate each of the update files being applied.

fn TEXT A HAS BEEN ERASED.

indicates that the text file to be produced does not have a filename of TEXT. File 'fn TEXT A' already existed. This prevents you from inadvertently including a down-level textfile in your nucleus.

ASMBLING fn

indicates that the assembly is going to begin. If you specified any assembler options on the VMFASM command line, the options used are also displayed.

fn {TEXT
TXTxxxxx} A1 CREATED

indicates the filename and filetype of the text deck.

Error Messages

```
DMSSTT002E FILE 'fn ASSEMBLE *' NOT FOUND
     The source file could not be located.

DMSSTT002E FILE 'ctlfile CNTRL *' NOT FOUND
     The control file could not be located.

*** ERROR UPDATING fn ***
     A return code from the UPDATE command was greater than
     12 but not equal to 40.  This means that a severe error
     occurred.  VMFASM terminates, and the return code
     from UPDATE is returned.

*** ERROR ASMBLING fn ***
     An assembler error occurred.

*** fn {TEXT     } A1 IS NOT CREATED ***
       {TXTxxxxx }
     No text file was produced because of assembler errors.
     VMFASM terminates with the return code from the ASSEMBLE
     command.

fn {TEXT     } A WAS NOT CREATED, RC=rc FROM COPYFILE
   {TXTxxxxx }
     The COPYFILE command failed to merge the fn UPDATES A1
     files or your A-disk is full.  The text file, therefore,
     failed with the return code shown.  VMFASM terminates
     with the return code from the COPYFILE command.
```

The format of the VMFPLC2 command is:

```
VMFPLC2      DUMP   {fn} {ft} [fm]      [(optionA optionB  ]
                    {* } {* } [* ]      [     optionD)     ]

             LOAD   [{fn} {ft} [fm]]    [(optionB optionC  ]
                    [{* } {* } [A ]]    [ optionD optionE  ]
                    [              ]    [  optionF)        ]

             SCAN   [{fn} {ft}]         [(optionB optionC  ]
                    [{* } {* }]         [ optionD optionF) ]

             SKIP   [{fn}{ft}]          [(optionB optionC  ]
                    [{* }{* }]          [     optionD)     ]

             MODESET          [(optionD)]

             tapcmd   [n]     [(optionD)]
                      [1]

          optionA:  [WTM  ]
                    [NOWTM]

          optionB:  [NOPRint]
                    [PRint  ]
                    [Term   ]
                    [DISK   ]
                    [APPend ]

          optionC:  [EOT  ]
                    [EOF n]
                    [EOF 1]

          optionD:  [cuu]      [DEN den]
                    [181]

          optionE:  [SELect]  [STOP]

          optionF:  [DATE]
```

where:

```
DUMP {fn}{ft}[fm]
     {* }{* }[* ]
```

dumps one or more disk files to tape. If fn and/or ft is specified as an asterisk (*) all files that satisfy the other file identifier are dumped.

If fm is coded as a letter, that disk and its extensions are searched for the specified file(s). If fm is coded as a letter and number, only files with that mode number and letter (and the extensions of the disk referenced by that fm letter) are dumped. If fm is coded as asterisk (*), all accessed disks are searched for the specified file(s). If fm is not specified, only the A-disk and its extensions are searched.

```
LOAD [ ⎧fn⎫ ⎧ft⎫ [fm]]
     [ ⎩* ⎭ ⎩* ⎭ [A ]]
```

reads tape files onto disk. If a file identifier is specified, only that one file is loaded. If the option EOF n is specified and no file identifier is entered, n tape files are written to disk. If an asterisk (*) is specified for fn or ft, all files within EOF n that satisfy the other file identifier are loaded.

The files are written to the disk indicated by the filemode letter. The filemode number, if entered, indicates that only files with that filemode number are to be loaded.

```
SCAN [ ⎧fn⎫ ⎧ft⎫ ]
     [ ⎩* ⎭ ⎩* ⎭ ]
```

positions the tape at a specified point, and lists the identifiers of the files it scans. Scanning occurs over n tape marks, as specified by the option EOF n (the default is 1 tape file). However, if a file identifier (fn and ft) is specified, scanning stops upon encountering that file. The tape is in a position to load the specified file.

```
SKIP [ ⎧fn⎫ ⎧ft⎫ ]
     [ ⎩* ⎭ ⎩* ⎭ ]
```

positions the tape at a specified point and lists the identifiers of the files it skips. Skipping occurs over n tape marks, as specified by the option EOF n (the default is 1 tape mark). However, if a file identifier (fn and ft) is specified, skipping stops after encountering that file; the tape remains positioned immediately following the file.

`MODESET` sets the values specified by the DEN, TRACK, and TRTCH options. After initial specification in a VMFPLC2 command, these values remain in effect for the virtual tape device until they are changed in a subsequent VMFPLC2 command, RDTAPE, WRTAPE, or TAPECTL macro.

```
tapcmd ┌ ┐
       │n│
       │1│
       └ ┘
```

specifies a tape control function (tapcmd) to be executed n times. Default is 1 if n is not specified. These functions also work on tapes in a non-CMS format.

Tapcmd	Action
BSF	Backspace n tape marks
BSR	Backspace n tape records
ERG	Erase gap
FSF	Forward-space n tape marks
FSR	Forward-space n tape records
REW	Rewind tape to load point
RUN	Rewind tape and unload
WTM	Write n tape marks

Options:

If conflicting options are specified, the last one entered is in effect.

`WTM` writes a tape mark on the tape after each file is dumped.

`NOWTM` writes a tape mark after each file is dumped, then backspaces over the tape mark so that subsequent files written on the tape are not separated by tape marks.

`NOPRINT` does not spool the list of files dumped, loaded, scanned, or skipped to the printer.

`PRINT` spools the list of files dumped, loaded, scanned, or skipped to the printer.

`TERM` displays a list of files dumped, loaded, scanned, or skipped at the terminal.

`DISK` creates a disk file containing the list of files dumped, loaded, scanned, or skipped. The disk file has the file identification of TAPE MAP A5.

`APPEND` causes the disk file (containing the list of files dumped, loaded, scanned, or skipped), to be added to the end of an existing TAPE MAP.

`EOT` reads the tape until an end-of-tape indication is received.

`EOF n` reads the tape through a maximum of *n* tape marks. The default is EOF 1.

`cuu`
`18n` specifies the virtual device address of the tape to be read or written to where n is 1, 2, 3, or 4. The default is 181. The unit specified by cuu must previously have been attached to your CMS virtual machine before any tape I/O operation can be attempted. Only virtual device addresses 181 through 184 are supported.

`DEN den` is the tape density where den is 200, 556, 800, 1600, or 6250. If 200 or 556 is specified, 7TRACK is assumed. If 1600 or 6250 is specified, 9TRACK is assumed; if 800 is specified, 9TRACK is assumed unless 7TRACK is specified. In the case of either 800/1600 or 1600/6250 dual-density drives, 1600 is the default if the 9TRACK option is specified. If neither the 9TRACK option nor the DEN option is specified, the drive operates at whatever bpi the tape drive was last set.

SELECT inhibits loading of a file from the tape that causes replacement of an *identical* file on the disk. Files will be loaded only if they do not exist on the specified disk, or when the date/time stamp for the file on the disk does *not match* the date/time stamp for the corresponding file on the tape.

STOP assumes that files contained on the tape are in alphabetical sequence. If the requested file is on the tape, the file is loaded onto disk and the tape stops. If the file is not on the tape and a file is encountered that is alphabetically beyond the bounds of the requested file, the tape stops. You must specify 'fn ft'. Neither 'fn' nor 'ft' may be specified as '*'.

DATE displays listfile information during a SCAN. The information displayed includes number of records, length of records, and date/time stamp.

Usage Notes

1. Tape records written by the VMFPLC2 command are 4005 bytes long. The first character is a binary 2 (X'02'), followed by the characters CMS and a file format byte, followed by 4000 bytes of file data packed without regard for logical record length. If a null block is dumped, the character "0" replaces the byte after CMS. This causes subsequent loading of null blocks to be ignored. In the final record, the character N replaces the blank after CMS, and the data area contains CMS file directory information.

2. If a tape file contains more CMS files than would fit on a disk, the tape load operation may terminate if there is not enough disk space to hold the files. To prevent this, when you dump the files, separate logical files by tape marks, then forward space to the appropriate file.

3. The CMS file directory is the first record of each CMS file on tape.

4. It is possible to run a tape off the reel in at least one situation. If you specify EOF *n* and *n* is greater than the number of tape marks on the tape, the tape will run off the reel.

5. The options for the 8809 tape drive must be 9TRACK and DEN 1600. Note that these are the default values, so you do not need to specify them.

6. When the STOP option is used with VMFPLC2 LOAD and the file does not exist, a return code of 44 is issued with message DMSVMF002E FILE 'fn ft fm' NOT FOUND.

7. For more information on tape file handling, see the *VM/SP CMS User's Guide*.

Other Messages and Codes

```
DMSVMF002E FILE 'fn ft fm' NOT FOUND  RC=28 and 44
DMSVMF003E INVALID OPTION 'option'  RC=24
DMSVMF010E PREMATURE EOF ON FILE NUMBER 'nn'  RC=40
DMSVMF014E INVALID FUNCTION 'function'  RC=24
DMSVMF017E INVALID DEVICE ADDRESS'cuu'  RC=24
DMSVMF023E NO FILETYPE SPECIFIED  RC=24
DMSVMF027E INVALID DEVICE 'device name'  RC=24
DMSVMF029E INVALID PARAMETER 'parameter' IN THE OPTION 'option' FIELD  RC=24
```

```
DMSVMF037E DISK 'mode' IS READ/ONLY  RC=36
DMSVMF042E NO FILEID SPECIFIED  RC=24
DMSVMF043E 'TAPn(cuu)' IS FILE PROTECTED  RC=36
DMSVMF047E NO FUNCTION SPECIFIED  RC=24
DMSVMF048E INVALID MODE 'mode'  RC=24
DMSVMF057E INVALID RECORD FORMAT  RC=32
DMSVMF058E END-OF-FILE OR END-OF-TAPE  RC=40
DMSVMF070E INVALID PARAMETER 'parameter'  RC=24
DMSVMF104S ERROR 'nn' READING FILE 'fn ft fm' FROM DISK  RC=100
DMSVMF105S ERROR 'nn' WRITING FILE 'fn ft fm' ON DISK  RC=100
DMSVMF110S ERROR READING 'TAPn(cuu)'  RC=100
DMSVMF111S ERROR WRITING 'TAPn(cuu)'  RC=100
DMSVMF113S TAPn(cuu) NOT ATTACHED  RC=100
DMSVMF115S {CONVERSION|{7|9|18}-TRACK|{800|6250}
           BPI|TRANSLATION|DUAL DENSITY|TRANSFER}
           FEATURE NOT SUPPORTED ON DEVICE 'cuu'  RC=88
DMSVMF613E VMFPLC2 MUST BE INVOKED AS A NUCLEUS EXTENSION  RC=40
DMSVMF701I NULL FILE  RC=
```

This page intentionally left blank.

Whenever a PTF is applied to a System Product Interpreter source program, the EXECUPDT command is used to create an executable version of the program. This command can only be used with System Product Interpreter programs. The following table indicates the options which should be selected in each case:

Change in EXEC	EXEC Procedure To Use	Indicated Options
ALL $XEDIT	EXECUPDT	CTL fn HIST SID
CANCEL $XEDIT	EXECUPDT	CTL fn HIST SID
CAPPEND $XEDIT	EXECUPDT	CTL fn HIST SID
EXECUPDT $EXEC	EXECUPDT	CTL fn HIST SID
EXECUTE $XEDIT	EXECUPDT	CTL fn HIST SID
PREFIXX $XEDIT	EXECUPDT	CTL fn HIST SID
PRFSHIFT $XEDIT	EXECUPDT	CTL fn HIST SID
PRFSHOW $XEDIT	EXECUPDT	CTL fn HIST SID
PROPHCHK $EXEC	EXECUPDT	CTL fn HIST
PROPLGER $EXEC	EXECUPDT	CTL fn HIST
PROPPCHK $EXEC	EXECUPDT	CTL fn HIST
PROPRTCV $EXEC	EXECUPDT	CTL fn HIST
PROPST $EXEC	EXECUPDT	CTL fn HIST
RGTLEFT $XEDIT	EXECUPDT	CTL fn HIST SID
SPLTJOIN $XEDIT	EXECUPDT	CTL fn HIST SID
STATUS $XEDIT	EXECUPDT	CTL fn HIST SID

"fn" references an UPDATE control file that controls the application of multiple UPDATE files to the source input file. See "Chapter 10. EXEC Procedures and Command Format Summaries" and the *VM/SP CMS Command and Macro Reference* for more information about the EXECUPDT command.

If you must regenerate the CMS nucleus, see "Updating CMS." If you must regenerate the CMSVSAM, CMSAMS, CMSDOS or CMSBAM segments, see "Chapter 5. Loading and Saving Discontiguous Saved Segments." All the other EXEC procedures for generating segments are described in "Chapter 10. EXEC Procedures and Command Format Summaries."

CMS and CMSL should be resaved whenever the system S-disk is updated. This will ensure that the S-disk directory, which is a part of the saved CMS nucleus, correctly matches the physical S-disk directory. See "Updating CMS."

See "Creating CMS Disk-Resident Modules," if a CMS module must be regenerated. A description of CMSGEND EXEC is presented in "CMSGEND" for further information.

If you apply a PTF to certain CP source programs, the corresponding CMS modules must also be regenerated to run properly. The source name, module name, and procedures used for regenerating the modules are shown in the following table[23].

Change in Source	Requires Regeneration of Module	EXEC Procedure(s) To Use
DMKDDR	DDR	GENERATE, CMSGEND
DMKDIR	DIRECT	GENERATE, CMSGEND
DMKFMT	None, no CMS execution	GENERATE
DMKNMT	IMAGELIB	CMSGEND
DMKIMG	GENIMAGE	CMSGEND
DMKRND	NCPDUMP	CMSGEND
DMSARN	ASM3705	INSTEP
DMSARD	ASM3705	INSTEP
DMSARX	ASM3705	INSTEP
DMSGRN	GEN3705	CMSGEND
DMSNCP	SAVENCP	CMSGEND
VMFDATE	VMFDATE	CMSGEND
VMFDOS	VMFDOS	CMSGEND
VMFLOAD	VMFLOAD	CMSGEND
VRSIZE	VRSIZE	CMSGEND
DMKTRR	TRAPRED	CMSGEND

See Appendix D, "Generating and Loading the 3704/3705 Control Program" for a description of the INSTEP EXEC procedure. All other EXEC procedures are described in "Chapter 10. EXEC Procedures and Command Format Summaries" if you need to examine them.

[23] Use this appendix only when applying updates and local fixes to your system. When corrective service is shipped via PUT, all modules are regenerated and replaced on a one-to-one basis.

Appendix D. Generating and Loading the 3704/3705 Control Program

Note: If you ordered EP/VS through the System Offering, use the INSTFPP EXEC as explained on page 130 to begin the installation process. Next, use the *Memo to Users* for EP/VS as reference and continue the installation on page 260 with Step 3B.

Several commands and EXEC procedures generate and load the 3704/3705 control program. These commands and EXEC procedures are executed in a CMS virtual machine. The commands are part of the VM/SP system and are distributed with it.

A special version of the IBM 3704/3705 Emulation Program Support Package for OS/VS, Order No. 5744-AN1 EP/VS SCP is available from PID for use under VM/SP. This version of the 3704/3705 package contains two CMS EXEC procedures for generating and loading the 3704/3705 control programs.

This chapter describes the step-by-step procedure for generating and loading the 3704/3705 control program. Each EXEC procedure and command is described as it is used. The action required at each step is summarized first and then explained in detail. "Part 1. Planning for System Generation" of the *VM/SP Planning Guide and Reference* lists all the documentation, physical devices, programming, and other materials you need before starting to generate the 3704/3705 control program.

Step 1. Log on the VM/SP System

VM/SP supports the EP type of control program. The VM/SP system that you load also must have been generated with:

- The IBM 3704 or 3705 Communications Controllers specified on a RDEVICE system generation macro.

- The NAMENCP macro coded to create an entry in the VM/SP system name table (DMKSNT) for the 3704/3705 control program.

- Space reserved on a CP-owned volume to contain a copy of the 3704/3705 control program.

These VM/SP system generation requirements are described in "Part 1. Planning for System Generation" of the *VM/SP Planning Guide and Reference.*

Step 2. Set Up a CMS Virtual Machine

You must IPL CMS in a virtual machine and be sure that OPT/LIBS the necessary devices are attached.

The 3704/3705 control program is generated using commands and EXEC procedures that execute in a CMS virtual machine. The CMS virtual machine must have the following resources:

- At least 1024K of virtual storage.[24]
- One tape drive (9 track, 800 or 1600 bpi).
- Space available on the CMS A-disk (120 cylinders of a 3330 disk, all 203 cylinders of a 2314 disk, 300 cylinders of a 3340 disk, 60 cylinders of a 3350 or 3380 disk, 80 cylinders of a 3375 disk, or 7200 pages of an FB-512 disk.

If the CMS virtual machine does not have these resources, use the CP DEFINE command to redefine the size of your virtual storage or send a message to the operator requesting the needed tape or disk device be attached.

Be sure that there are no files on the A-disk with a filetype of COPY or TEXT. Use the CMS RENAME command to temporarily change such filetypes. A naming conflict can terminate the installation procedure for the distribution tape.

You need CP command privilege classes A, B, or C to install the 3704/3705 control program. Check with the system administrator to ensure that your VM/SP directory entry has the appropriate command privileges.

Step 3A. Load the IBM 3704/3705 Control Program Distribution Tape Files onto a CMS Disk that has been accessed as filemode A.

Note: Do **not** access the CMS System-Disk (190) as filemode A.

Use CMS commands to position the distribution tape at the proper file and to create CMS disk files from the tape files. The first file created from the tape files is an EXEC procedure that processes the rest of the tape files and creates the CMS disk files.

Have the distribution tape mounted and attached to your virtual machine. The distribution tape contains ten physical files, separated by tape marks. The tenth file contains the INSTEP and ARNGEND EXEC procedures, which create the necessary CMS files from the other tape files.

Use the CMS TAPE command to position the tape at the beginning of the tenth file:

```
tape rew
tape fsf 9
```

[24] Depending on the size and options of the EP Program being built, and depending on the blocksize and track size of DASD used, the minimum amount of virtual storage might have to be increased. Verify that the 3705 assembler has been built; then use a larger virtual storage size, e.g., 4096K or greater to continue.

Then, use the CMS TAPPDS command to create the INSTEP EXEC A1 and ARNGEND EXEC A1 files from the tenth file:

```
tappds * exec
```

If the files are successfully created, the responses

```
FILE 'ARNGEND EXEC A1' COPIED
FILE 'DMSARD  EXEC A1' COPIED[25]
FILE 'DMSARX  EXEC A1' COPIED[25]
FILE 'DMSGRN  EXEC A1' COPIED[25]
FILE 'DMSTMA  EXEC A1' COPIED[25]
FILE 'INSTEP  EXEC A1' COPIED
```

appear on your terminal. Before invoking the INSTEP EXEC procedure do the following:

1. XEDIT INSTEP EXEC, deleting the RENAME commands for DMSARD, DMSARX, DMSGRN, and DMSTMA.

2. Erase the DMSARN, DMSARX, DMSGRN, and DMSTMA EXEC files.

3. Access the CMS System-Disk (190) as T/A to get access to the mode 1 files.

Invoke the INSTEP EXEC procedure to load all the necessary files and generate the 3705 Assembler:

```
instep
```

The INSTEP EXEC procedure generates the 3705 Assembler and creates the macro and text libraries that are needed to generate a 3704/3705 control program. The INSTEP EXEC procedure sends messages to the terminal to indicate its progress.

INSTEP issues the message:

```
BUILD STAGE ONE MACLIB
LOADING 'GEN3705 MACLIB'
```

and uses the third tape file to create the CMS file GEN3705 MACLIB A1. INSTEP also issues the messages:

```
BUILD STAGE TWO MACLIBS
LOADING 'MAC3705 MACLIB'
```

using the fifth tape file to create the CMS file MAC3705 MACLIB A1. Using the sixth tape file, INSTEP creates the CMS file OBJ3705 MACLIB A1, and issues the messages:

```
BUILD STAGE TWO TXTLIB

LOADING 'OBJ3705 MACLIB'
RENAME OBJ3705 MACLIB A1 OBJ3705 TXTLIB A1
```

[25] These files may not be present on later releases of the EP/VS distribution tape.

Finally, INSTEP issues the message:

```
LOAD 3705 ASSEMBLR FILES
```

and loads the assembler text files from tape via the TAPPDS command. The files copied are listed off in messages in the form:

```
FILE 'fn EPTAPE A1' COPIED
```

The ARNGEND EXEC procedure is invoked by INSTEP to generate the 3705 Assembler, after issuing the message:

```
BUILD 3705 ASSEMBLR MODULES.
```

Step 3B. Invoking ARNGEND EXEC

Note: System Offering users who have entered at this step must manually execute the ARNGEND EXEC before proceeding.

The ARNGEND EXEC procedure displays the following status and error messages:

```
ENTER TARGET DISK MODE FOR 3705 ASSEMBLR MODULES
DEFAULTS TO S-DISK IF NONE ENTERED
```

You enter the mode letter of the disk that will contain the 3705 assembler modules when the assembler is used. This may be a different disk then the one on which the modules now reside. If you enter a mode letter, ARNGEND uses that mode letter as the "targetmode" operand of the GENDIRT command when it creates the auxiliary directory for the 3705 assembler. If you do not specify a mode letter, S is assumed by the GENDIRT command.

If the 3705 assembler text files are not loaded successfully, or if the assembler generation procedure fails, the following message appears:

```
ASM3705 GEND FAILED
```

When the last message:

```
END OF EPTAPE INSTALL
```

appears on the terminal, the distribution tape is no longer needed. At this time, the 3705 Assembler program, the macro libraries for the Stage 1 and Stage 2 generation procedures, and the text library for the Stage 2 generation procedure all exist on the CMS A-disk.

Note: You may find it helpful to dump the contents of the A-disk to tape at this time. If you save the tape dump, you have the pre-Stage 1 files. If errors are later encountered, you may need these files.

Appendix H. Installation Reference Worksheet

Record important instructions

Instruction	Number	Address	Name (Label)
Mount volume	________________	as ____________	________________________
Mount volume	________________	as ____________	________________________
Mount volume	________________	as ____________	________________________
Mount volume	________________	as ____________	________________________
Mount tape	________________	as ____________	________________________
Mount tape	________________	as ____________	________________________

Record device addresses and device types

Label	Address	Device Type
VMSRES		
VMPK01		
VMSEXT		
VMSTGE		
Storage diskpack VMSTG2		

Record other device addresses and device types

Device	Address	Device Type
Console		
First Tape Drive		
Second Tape Drive		
Printer		
Punch		
Reader		
Graphics		

PRODUCT	ABBREV	NUMBER	LEVEL	PREREQ
VS APL	VSAPL	5748-AP1	1.4.0	5748-XXH[5]
VS FORTRAN	FORTRAN	5748-FO3	1.3.0	

Figure 25 (Part 3 of 3). VM/SP Optional Feature Program Products

[1] EREP is ordered as a feature of the SCP 5749-010.

[2] HPO is ordered separately as Program Product number 5664-173. If you are installing VM/SP HPO with the merged product tape, use the *VM/SP HPO Installation Guide* instead of this manual for installation.

[3] The VM/VSE Data Feature is available as Program Product number 5668-919. The MVS Data Feature is available as Program Product number 5665-955. Ordering information for either of these products can be found in the latest announcement letters for these products. The System Offering registration process is not used for ordering these two products.

[4] PROFS, Program Product number 5664-176, installation is supported. To order PROFS and the associated documentation, refer to the latest announcement letter. The System Offering registration process is not used for ordering PROFS.

[5] Required only when using the VS APL Session Manager (AP126).

Reference Manuals

There are reference manuals for all of the Feature Products in the above chart. Refer to the VM/SP 3.1 ordering instructions for the list of available feature program product manuals. Your IBM representative or nearest IBM branch office can help you obtain these publications and can tell you how to subscribe to them so you automatically get updates and new editions.

DASD Requirements

Before you install your program products, you will want to ensure that MAINT 319 (the P-disk) and MAINT 325 (the W-disk) are large enough for each DASD type as determined by the combinations of products you choose to install. Figure 26 shows the number of 1K blocks required on the MAINT 319 and MAINT 325 minidisks for each program product. Figure 27 shows the 1K sizes of the MAINT 319 and MAINT 325 minidisks as defined by the sample directories for each DASD type. If your total of 1K blocks needed exceeds the space allocated in the sample directories, you will need to update your directory with larger sizes for the MAINT 319 and MAINT 325 minidisks as required.

Program Product	Product Unique Userid	319	325
ADRS		5	1,135
ADRS BG		5	2,285
APL/DI		5	1,455
BATCH	VMBATCH VMBATCH1	75	
COBOL		2,450	
COBOL DEBUG		850	
CSP/AD	CSPUSER	15	
CSP/AE	CSPUSER	30	
DCF		1,860	
DIRMAINT	DIRMAINT DATAMOVR	485	
DMS/CMS		720	
FILE STORAGE	FSFTASK1 FSFCNTRL	270	
FORTRAN DEBUG		220	
EP/VS		5	
FPS		5	670
EREP		25	
GDDM/PGF/IMD		7,600	
IBM BASIC		15	
IIPS	IIPS	10	
INFO/SYS	ISMAINT	15	
IPCS/E	IPCS	725	

Figure 26 (Part 1 of 2). Feature Product DASD Utilization

SQLDBA 195	the SQL/DS production minidisk used for all SQL/DS operations.
SQLDBA 200	the directory disk for the SQL/DS starter data base.
SQLDBA 201	the log disk for the SQL/DS starter data base.
SQLDBA 202	the data disk for the SQL/DS starter data base.

- SQLUSER SQLUSER is a SQL/DS user machine used to to administrative tasks needed to complete installation.

  The SQLUSER SQLUSER's minidisk is used as follows:

SQLUSER 191	A-disk, work disk for the SQLUSER user machine.

Reclaiming DASD Space

- The sample directories contain entries for each of the supported feature product userids. You may want to tailor these directories for better utilization of DASD space after the installation is complete. For instance, space allocated for products not installed may be used as a different userid or deleted. Refer to Figure 26 for unique userids for each of the products.

  For example, if you did not install the FILE STORAGE program product, you can xedit the directory and remove the entries for userids FSFTASK1 and FSFCNTRL thereby reclaiming the minidisk space (cylinders [CKD] or blocks [FBA]) set aside for its unique disks that you would not be using.

  Once you have made all your changes, use the CMS DIRECT command to process the file to see if it follows the required directory format.

  ```
  DIRECT VMUSERS DIRECT (EDIT
  ```

  Notes:

  1. If DIRMAINT is installed, refer to the DIRMAINT documentation for directions for updating the directory.
  2. You can also use the DISKMAP EXEC to produce a directory map.

- The installation procedures of Optional Feature Program Products do not use the performance advantage of the system Y-disk (19E). All program product files moved to this minidisk, which are to be included in the Y-stat, must have filemode 2.

  Note: Any changes to the Y-disk require that you resave CMS and CMSL.

  Some products, such as COBOL and PL/1, require further work to establish a new auxiliary directory. If you get the DMSACC723I error message when IPLing CMS, refer to Appendix F, "Example of Additional CMS Segment Containing Y-STATS" on page 279 for directions about adding CMS segments.

VMSRES Pack Layout for 3310 Starter System

FB-512 Block Number	No. of FB-512 Blocks	Page Number	Number of Pages (BLK/8)	Contents
0-15	16	0-1	2	Allocation
16-9375	9360	2-1171	1170	Saved segments
9376-14657	5282	1172-	**	191 minidisk for MAINT
14658-35177	20520		**	194 minidisk for MAINT (CP)
35178- 36201	1024		**	191 minidisk for OPERATOR
36202-37113	912		**	191 minidisk for EREP
37114-37569	456		**	191 minidisk for OPERATNS
37570-46911	9342	-5863	**	xxx minidisk for user option
46912-92479	45568	5864-11559	5696	190 MAINT (CMS system minidisk)*
92480-94303	1824	11560-11787	228	VM/SP directory
94304-94815	512	11788-11851	64	Spool file checkpoint
94816-115715	20900	11852-	**	19D minidisk for MAINT (HELP)
115716-122199	6484	-15274	**	Spool space
122200-122711	512	15275-15338	64	Warm start data
122712-123223	512	15339-15402	64	I/O Error Recording area
123224-123479	256	15403-15434	32	191 minidisk for IVPM1
123480-123735	256	15435-15466	32	191 minidisk for IVPM2
123736-126015	2280	15467-15751	285	CP nucleus

Figure 34. VMSRES Layout for a 3310 Starter System

* The CMS nucleus must start on a 256 block boundary. 43520 displacement in the minidisk will contain the CMS nucleus. The CMSL nucleus must start on a 256 block boundary. 44544 displacement in the minidisk will contain the CMSL nucleus.

** Not on a page boundary.

3310 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

3310 DASD are FB-512 devices and the allocations are expressed as pages.

VMSRES	
perm	00002 11559
temp	----- -----
drct	11560 11787
tdsk	----- -----
perm	11788 14464
temp	14465 15274
perm	15275 15751
temp	
perm	
tdsk	
perm	
end	

VMPK01	
perm	00002 03421
tdsk	----- -----
temp	03422 05921
tdsk	----- -----
perm	05922 11374
tdsk	11375 15751
perm	
end	

VMSEXT	
perm	00002 10030
tdsk	----- -----
perm	----- -----
temp	10031 12530
perm	12531 15751
perm	
end	

VMSTGE	
perm	00002 15751

Figure 34.1. VMSRES, VMPK01, VMSEXT, and VMSTGE Sample Allocations for 3310

```
********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 10, PAGE 22 TO CYL 14, PAGE 34 (241 PAGES)
********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(10,22),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 14, PAGE 35 TO CYL 22, PAGE 27 (449 PAGES)
********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(014,35),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*    CYL 1 (PAGE 01) TO CYL 4 (PAGE 022)
*    TOTAL 193 PAGES
********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 39 (Part 8 of 8). Sample DMKSNT File for a 3330 Device

VMSRES Pack Layout for 3330 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-20	20	Reserved for saved segments
21-32	12	191 minidisk for MAINT
33-77	45	194 minidisk for MAINT (CP)
78-97	20	xxx minidisk for user option
98-197	100	190 minidisk for MAINT (CMS system disk)*
198-199	2	Spool file checkpoint
200-274	75	Page/spool space
275-278	4	VM/SP directory
279-280	2	191 minidisk for OPERATOR
281-282	2	191 minidisk for EREP
283-299	17	xxx minidisk for user option
300	1	191 minidisk for OPERATNS
301-316	16	193 minidisk for OPERATNS
317-330	14	xxx minidisk for user option
331-390	60	19D minidisk for MAINT (HELP)
391-392	2	Warm start data
393-394	2	I/O Error Recording area
395	1	191 minidisk for IVPM1 user
396	1	191 minidisk for IVPM2 user
397-403	7	CP nucleus

Figure 40. VMSRES Layout for a 3330 Starter System

* Specify cylinder 94 to load the CMS nucleus which occupies the first 3 cylinders of the recompiled area.
Specify cylinder 97 to load the CMSL nucleus which occupies the last 3 cylinders of the recompiled area.

VMSRES	
perm	000 199
temp	200 274
drct	275 278
tdsk	--- ---
perm	279-403
temp	--- ---
perm	279 807
temp	
perm	
tdsk	
perm	
end	

VMPK01	
perm	000 119
tdsk	120 214
temp	215 334
tdsk	--- ---
perm	335 403
tdsk	--- ---
perm	335 807
end	

VMSEXT	
perm	000 000
tdsk	001 110
perm	111 296
temp	297 391
perm	392 403
perm	392 807
end	

VMSTGE	
perm	000 403
	000 807

Figure 40.1. VMSRES, VMPK01, VMSEXT, and VMSGTE Sample Allocations for 3330

This page intentionally left blank.

VMSRES Pack Layout for 3340-70 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-28	28	Reserved for saved segments
29-31	3	not used
32-43	12	CP nucleus
44-69	26	191 minidisk for MAINT
70-79	10	VM/SP directory
80-84	5	I/O Error Recording area
85-89	5	Warm start data
90-94	5	Spool file checkpoint
95-124	30	Page/spool space
125-360	236	190 minidisk for MAINT (CMS system disk)*
361-362	2	191 minidisk for OPERATOR
363-366	4	191 minidisk for EREP
367-488	122	xxx minidisk for user option
489-588	100	19D minidisk for MAINT (HELP)
589-695	107	194 minidisk for MAINT (CP)

Figure 46. VMSRES Layout for a 3340-70 Starter System

* Specify cylinder 222 to load the CMS nucleus which occupies the first 7 cylinders of the recompiled area.
Specify cylinder 229 to load the CMSL nucleus which occupies the last 7 cylinders of the recompiled area.

3340-70 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

All these allocations are expressed as cylinders.

VMSRES	
perm	000 069
temp	--- ---
drct	070 079
tdsk	--- ---
perm	080 094
temp	095 124
perm	125 695
temp	
perm	
tdsk	
perm	
end	

VMPK01	
perm	000 285
tdsk	--- ---
temp	286 366
tdsk	367 616
perm	617 695
tdsk	--- ---
perm	
end	

VMSEXT	
perm	000 464
temp	465 625
perm	626 695
temp	
perm	
perm	
end	

VMSTGE	
perm	000 695

Figure 46.1. VMSRES, VMPK01, VMSEXT, and VMSTGE Sample Allocations for 3340-70

```
********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 5, PAGE 55 TO CYL 7, PAGE 55 (241 PAGES)
********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(05,55),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 7, PAGE 56 TO CYL 11, PAGE 24 (449 PAGES)
********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(007,56),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE 73  (193 PAGES)
********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 51 (Part 8 of 8). Sample DMKSNT File for a 3350 Device

VMSRES Pack Layout for 3350 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-17	7	191 minidisk for MAINT
18-39	22	194 minidisk for MAINT (CP)
40	1	191 minidisk for OPERATOR
41-51	11	xxx minidisk for user option
52-100	49	190 minidisk for MAINT (CMS system disk)*
101-102	2	VM/SP directory
103-116	14	xxx minidisk for user option
117	1	191 minidisk for OPERATNS
118-125	8	193 minidisk for OPERATNS
126-148	23	xxx minidisk for user option
149-193	45	319 minidisk for MAINT
194-221	28	19E minidisk for MAINT
222-233	12	xxx minidisk for user option
234-276	43	Spool space
277	1	Spool file checkpoint
278-317	40	Spool space
318-417	100	T-DISK space
418-445	28	19D minidisk for MAINT (HELP)
446	1	191 minidisk for EREP
447-468	22	201 minidisk for MAINT
469-480	12	293 minidisk for MAINT (CMS service)
481-492	12	294 minidisk for MAINT (CP service)
493-542	50	xxx minidisk for user option
543	1	191 minidisk for IVPM1
544	1	191 minidisk for IVPM2
545-546	2	Warm start data
547-548	2	I/O Error Recording area
549-554	6	CP nucleus

Figure 52. VMSRES Layout for a 3350 Starter System

* Specify cylinder 45 to load the CMS nucleus which occupies the first 2 cylinders of the recompiled area.
Specify cylinder 47 to load the CMSL nucleus which occupies the last 2 cylinders of the recompiled area.

3350 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

All these allocations are expressed as cylinders.

VMSRES	
perm	000 100
temp	--- ---
drct	101 102
tdsk	--- ---
perm	103 233
temp	234 276
perm	277 277
temp	278 317
perm	--- ---
tdsk	318 417
perm	418 554
end	

VMPK01	
perm	000 226
tdsk	--- ---
temp	227 326
tdsk	
perm	327 554
tdsk	
perm	
end	

VMSTGE	
perm	000 554

Figure 52.1. VMSRES, VMPK01, and VMSTGE Sample Allocations for 3350

This page intentionally left blank.

```
*********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 4288 (PAGE 536) TO BLK 6215 (PAGE 776)
*     TOTAL = 241 PAGES
*********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(536),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*       FB-512 BLK 6216 (PAGE 777) TO BLK 9807 (PAGE 1225)
*     TOTAL = 449 PAGES
*********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(777),                                          X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
*********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*   THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*       FB-512 BLK 16 (PAGE 2) TO BLK 1559 (PAGE 194)
*     TOTAL = 193 PAGES
*********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(2),                                            X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 57 (Part 8 of 8). Sample DMKSNT File for a 3370 Device

VMSRES Pack Layout for 3370 Starter System

Block No. (Directory use)	No. of FB-512 Blocks	Page Number (CP Format use)	Number of Pages (BLK/8)	Contents
0-15	16	0-1	2	Allocation
16-9375	9360	2-1171	1170	Saved segments
9376-14657	5282	1172-	**	191 minidisk for MAINT
14658-35177	20520		**	194 minidisk for MAINT (CP)
35178-36201	1024		**	191 minidisk for OPERATOR
36202-46911	10710	-5863	**	xxx minidisk space for user option
46912-92479	45568	5864-11559	5696	190 MAINT (CMS system minidisk) *
92480-94303	1824	11560-11787	228	VM/SP directory
94304-94559	256	11788-	32	191 minidisk for IVPM1
94560-94815	256	11820-	32	191 minidisk for IVPM2
94816-122199	27384	11852-15274	3423	Spool space
122200-122711	512	15275-15338	64	Warm start data
122712-123223	512	15339-15402	64	I/O Error Recording area
123224-123735	512	15403-15466	64	Spool file checkpoint
123736-126015	2280	15467-15751	285	CP nucleus
126016-276015	150000	15752-36472	18750	Spool space
276016-276927	912	36471-45856	114	191 minidisk for EREP
366856-367311	456	45857-45913	57	191 minidisk for OPERATNS
367312-374607	7296	45914-46825	912	193 minidisk for OPERATNS
374608-380327	5720	46826-47540	715	xxx minidisk space for user option
380328-415327	35000	47541-51915	4375	T-DISK space
415328-417607	2280	51916-52200	285	xxx minidisk space for user option
417608-438507	20900	52201-	**	19D minidisk for MAINT
438508-557999	119492		**	xxx minidisk space for user option

Figure 58. VMSRES Layout for a 3370 Starter System

* The CMS nucleus must start on a 256 block boundary. 43520 displacement in the minidisk will contain the CMS nucleus.
The CMSL nucleus must start on a 256 block boundary. 44544 displacement in the minidisk will contain the CMSL nucleus.

** Not on a page boundary.

3370 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

3370 DASD are FB-512 devices and the allocations are expressed as pages.

VMSRES	
perm	00002 11559
temp	----- -----
drct	11560 11787
tdsk	----- -----
perm	11788 11851
temp	11852 15274
perm	15275 15751
temp	15752 34501
perm	34502 47540
tdsk	47541 51915
perm	51916 69749
end	

VMPK01	
perm	00002 14458
tdsk	14459 26958
temp	26959 36958
tdsk	----- -----
perm	36959 69749
tdsk	----- -----
perm	
end	

VMSTGE	
perm	00002 69749

Figure 58.1. VMSRES, VMPK01, and VMSTGE Sample Allocations for 3370

This page intentionally left blank.

VMSRES Pack Layout for 3375 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-14	4	Not used
15-23	9	191 minidisk for MAINT
24-55	32	194 minidisk for MAINT (CP)
56-69	14	xxx minidisk space for user option
70-143	74	190 minidisk for MAINT (CMS system disk)*
144-147	4	VM/SP directory
148-287	140	T-DISK space
288-478	191	Spool space
479-480	2	Spool file checkpoint
481-555	75	Spool space
556	1	191 minidisk for OPERATNS
557-568	12	193 minidisk for OPERATNS
569-633	65	319 minidisk for MAINT
634-675	42	19E minidisk for MAINT
676-695	20	xxx minidisk for user option
696-737	42	19D minidisk for MAINT (HELP)
738-909	172	xxx minidisk for user option
910-911	2	191 minidisk for OPERATOR
912-934	23	xxx minidisk for user option
935-936	2	191 minidisk for EREP
937-946	10	xxx minidisk for user option
947-948	2	Warm start data
949-950	2	I/O Error Recording area
951	1	191 minidisk for IVPM1
952	1	191 minidisk for IVPM2
953-958	6	CP nucleus

Figure 64. VMSRES Layout for a 3375 Starter System

* Specify cylinder 70 to load the CMS nucleus which occupies the first 2 cylinders of the recompiled area.
Specify cylinder 72 to load the CMSL nucleus which occupies the last 2 cylinders of the recompiled area.

3375 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

All these allocations are expressed as cylinders.

VMSRES	
perm	000 143
temp	--- ---
drct	144 147
tdsk	148 287
perm	--- ---
temp	288 478
perm	479 480
temp	481 555
perm	556 958
tdsk	
perm	
end	

VMPK01	
perm	000 226
tdsk	--- ---
temp	227 326
tdsk	
perm	327 958
tdsk	
perm	
end	

VMSTGE	
perm	000 958

Figure 64.1. VMSRES, VMPK01, and VMSTGE Sample Allocations for 3375

```
**********************************************************************
*  VS/FORTRAN (PROGRAM NO. 5748-F03)
*     HEX LOAD ADDRESS FOR SEGMENT 113 = 710000
*     THE SPACE FOR DSSVFORT IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 4, PAGE 85 TO CYL 6, PAGE  25 (241 PAGES)
**********************************************************************
         SPACE
DSSVFORT NAMESYS SYSNAME=DSSVFORT,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=960K,                                           X
               SYSHRSG=(113,114,115,116,117,118,119,120,121,122,123,124X
               ,125,126,127),                                          X
               SYSPGNM=(1808-2047),                                    X
               SYSPGCT=240,                                            X
               SYSSTRT=(04,85),                                        X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
**********************************************************************
*  GDDM/PGF (PROGRAM NO. 5748-XXH)
*     HEX LOAD ADDRESS FOR SEGMENT 146 = 920000
*     THE SPACE FOR ADMASSSV IS ALLOCATED ON VMPK01 AS FOLLOWS:
*          CYL 6, PAGE  26 TO CYL 9, PAGE  24 (449 PAGES)
**********************************************************************
         SPACE
ADMASS30 NAMESYS SYSNAME=ADMASS30,                                     X
               SYSVOL=VMPK01,                                          X
               SYSSIZE=1024K,                                          X
               SYSHRSG=(146,147,148,149,150,151,152,153,154,155,156,157X
               ,158,159,160,161,162,163,164,165,166,167,168,169,170,171X
               ,172,173),                                              X
               SYSPGNM=(2336-2783),                                    X
               SYSPGCT=448,                                            X
               SYSSTRT=(06,026),                                       X
               VSYSRES=,                                               X
               VSYSADR=IGNORE
         EJECT
**********************************************************************
*  PROFS    (PROGRAM NO. 5799-BEX)
*     HEX LOAD ADDRESS FOR SEGMENT  83 = 530000
*  THE SPACE FOR OFSSEG IS ALLOCATED ON PROFPK, AS FOLLOWS:
*          CYL 1, PAGE 1 TO CYL 2, PAGE  43 (193 PAGES)
**********************************************************************
         SPACE
PROFS    NAMESYS SYSNAME=OFSSEG,                                       X
               SYSSIZE=384K,                                           X
               VSYSRES=,                                               X
               VSYSADR=IGNORE,                                         X
               SYSCYL=,                                                X
               SYSVOL=PROFPK,                                          X
               SYSSTRT=(01,01),                                        X
               SYSPGCT=192,                                            X
               SYSPGNM=(1328-1519),                                    X
               SYSHRSG=(83,84,85,86,87,88,89,90,91,92)
         EJECT
         END
```

Figure 69 (Part 8 of 8). Sample DMKSNT File for a 3380 Device

VMSRES Pack Layout for 3380 Starter System

Real Cylinder Address	Number of Cylinders	Contents
0	1	Allocation
1-10	10	Reserved for saved segments
11-18	8	191 minidisk for MAINT
19-39	21	194 minidisk for MAINT (CP)
40	1	191 minidisk for OPERATOR
41-51	11	xxx minidisk for user option
52-96	45	190 minidisk for MAINT (CMS system disk)*
97-98	2	VM/SP directory
99	1	191 minidisk for EREP
100-126	27	19E minidisk for MAINT
127-171	45	319 minidisk for MAINT
172-211	40	xxx minidisk for user option
212	1	191 minidisk for OPERATNS
213-220	8	193 minidisk for OPERATNS
221-247	27	19D minidisk for MAINT (HELP)
248-401	154	xxx minidisk for user option
402-441	40	Spool space
442	1	Spool file checkpoint
443-482	40	Spool space
483-582	100	T-DISK space
583-831	249	xxx minidisk for user option
832-852	21	201 minidisk for MAINT
853-863	11	293 minidisk for MAINT (CMS service)
864-874	11	294 minidisk for MAINT (CP service)
875	1	191 minidisk for IVPM1
876	1	191 minidisk for IVPM2
877-878	2	Warm start data
879-880	2	I/O Error Recording area
881-884	4	CP nucleus

Figure 70. VMSRES Layout for a 3380 Starter System

* Specify cylinder 43 to load the CMS nucleus which occupies the first 1 cylinder of the recompiled area.
Specify cylinder 44 to load the CMSL nucleus which occupies the last 1 cylinder of the recompiled area.

3380 Sample Allocations

These tables show the individual pack sample allocations after base installation is complete.

All these allocations are expressed as cylinders.

VMSRES	
perm	000 096
temp	--- ---
drct	097 098
tdsk	--- ---
perm	099 401
temp	402 441
perm	442 442
temp	443 482
perm	--- ---
tdsk	483 582
perm	583 884
end	

VMPK01	
perm	000 326
tdsk	--- ---
temp	327 426
tdsk	
perm	427 884
tdsk	
perm	
end	

VMSTGE	
perm	000 884

Figure 70.1. VMSRES, VMPK01, and VMSTGE Sample Allocations for 3380

This page intentionally left blank.

SC24-5237-1

VM/SP Installation Guide (File No. S370/4300-34) Printed in U.S.A. SC24-5237-1

SC24-5237-1

VM/SP Installation Guide (File No. S370/4300-34) Printed in U.S.A. SC24-5237-1