Prime®

System Administrator's
Guide
Volume I
System Configuration

Revision 22.0

DOC10131-2LA

System Administrator's Guide
Volume I
System Configuration

Second Edition

Peter A. Neilson

Printing History

Credits

How to Order Technical Documents

To order copies of documents, or to obtain a catalog and price list:

United States Customers

Call Prime Telemarketing,
toll free, at 1-800-343-2533,
Monday through Friday,
8:30 a.m. to 5:00 p.m. (EST).

International

Contact your local Prime
subsidiary or distributor.

Customer Support

Prime provides the following toll-free numbers for customers in the United States needing service:

1-800-322-2838 (Massachusetts)
1-800-541-8888 (Alaska and Hawaii)
1-800-343-2320 (within other states)

For other locations, contact your Prime representative.

Surveys and Correspondence

Please comment on this manual using the Reader Response Form provided in the back of this book. Address any additional comments on this or other Prime documents to:

Technical Publications Department
Prime Computer, Inc.
500 Old Connecticut Path
Framingham, MA   01701

WARNING

This equipment generates and uses radio frequency energy and if not installed and used properly, i.e., in strict accordance with the instructions manual, may cause harmful interference to radio communications. It has been tested and found to comply with the limits for a Class A computing device pursuant to Subpart J of Part 15 of FCC rules, which are designed to provide reasonable protection against such interference when operated in a commercial environment.

Operation of this equipment in a residential area is likely to cause interference, in which case the user at his own expense will be required to take whatever measures may be required to correct the interference.

If there are any questions, please contact your Prime Field Office.

CONTENTS

APPENDICES

ABOUT THIS BOOK

The *System Administrator's Guide, Volume I: System Configuration*, is the first volume of the System Administrator's Guide set. It helps you plan your configuration, and also mentions installation issues and the responsibilities of the System Administrator. If you have administrative responsibility for a Prime system, this book is intended for you. It contains a master dictionary of all configuration directives, an overview of all the PRIMOS® directories and files, and a template for the PRIMOS.COMI file. Other administrative functions are described in the remaining books in the set:

- *System Administrator's Guide, Volume II, Communication Lines and Controllers* (DOC10132-2LA) provides definitions, instructions, and examples of the directives and commands necessary to configure communication lines.

- *System Administrator's Guide, Volume III, System Access and Security* (DOC10133-2LA) documents all the security features available on the operating system, including ACLs, EDIT_PROFILE, and the Security Audit facility. It also describes environmental factors and orderly procedures necessary to maintain the security of terminals, peripherals, and storage media.

You are expected to have some familiarity with Prime systems before reading the volumes of the System Administrator's Guide. If you are not familiar with the PRIMOS operating system, read the *PRIMOS User's Guide* (DOC4130-5LA), which explains Prime's file management system and describes essential commands and utilities.

CONTENTS

This book contains:

Chapter 1, Overview: An explaination of the responsibilities of the System Administrator, lists of the the directories containing PRIMOS software, and lists of servers and other system services.

Chapter 2, Disks and Tape Drives: An explaination of the allocation of disk space, both for file storage and for paging, on all disk systems supported by Prime equipment. This chapter also explains how to control tape drive assignment.

Chapter 3, Initializing System Software: Information about ensuring the proper initialization of shared system software, including shared segments, library EPFs, and shared static-mode libraries.

Chapter 4, Adding and Modifying System Software: Information about adding and modifying system software.

Chapter 5, Planning the System Configuration: An explaination of the several categories of configuration directives, and how to use them.

Chapter 6, Configuration Directives: A reference to all the configuration directives.

Appendix A, Obsolete and Rarely Used Commands and Directives: Commands and directives that are obsolete or are best avoided, as an aid to updating your system to current usage.

RELATED DOCUMENTATION

Here is other Prime documentation that will be of help to you:

- *Rev. 22.0 Software Installation Guide* (IDR10176-2XA). This book describes how to install Rev. 22.0 PRIMOS, either on a new system, or when upgrading from earlier revisions of PRIMOS. It also contains sample startup and configuration files.

- *ICS User's Guide* (DOC10094-1LA) and its update package for Rev. 21.0 (UPD10094-11A) This book provides detailed information on Prime's Model 2 (ICS2) and Model 3 (ICS3) Intelligent Communications Subsystems.

- *Operator's System Overview* (DOC9298-3LA). This book introduces the series of operator's guides and describes computer-room operation of Prime systems.

- *Operator's Guide to System Monitoring* (DOC9299-3LA). This book describes how to monitor system activity and respond to system and user messages.

- *Operator's Guide to File System Maintenance* (DOC9300-4LA). This book describes the PRIMOS file system and explains how to format partitions with MAKE, how to run the disk maintenance program FIX_DISK, how to determine physical device numbers, and how to interpret disk error messages.

- *Data Backup and Recovery Guide* (DOC10129-1LA) and its update (UPD10129-11A). This book describes how to save information on disk or tape and how to restore that information later.

- *Operator's Guide to System Commands* (DOC9304-4LA). This book details the commands used by system operators.

● *Operator's Guide to the Spooler Subsystem* (DOC9303-3LA). This book describes how to set up, monitor, and control the Spooler subsystem.

● *PRIMOS Commands Reference Guide* (DOC3108-7LA). This book is a detailed reference of user commands.

● *Programmer's Guide to BIND and EPFs* (DOC8691-1LA) and its update package for Rev. 22.0, (UPD8691-11A). Administrators should read this book's information about EPF Libraries.

● *Site Preparation Guide* (DOC5029-3LA) its update packages (UPD5029-31A and UPD5029-32A), and the release notes (RLN5029-3LA). This book provides information for preparing and maintaining a system site.

● *DSM User's Guide* (DOC10061-2LA). This book describes Prime's Distributed Systems Management, and tells how to use DSM to log events, to monitor status, and to control operations, either on a single system or on a network of systems.

● *Remote Job Entry Phase II Guide* (DOC6053-4LA) and its update (UPD6053-41A). This book contains information on setting up the directories required for RJE.

● *Translator Family Software Release Document* (DOC10217-1PA). This book contains information on installation of compilers, assemblers and editors. It also explains how to change the default values of compiler options.

NETWORK DOCUMENTATION AVAILABLE AT REV. 22.0

● *PRIMENET Planning and Configuration Guide* (DOC7532-4LA)

● *Programmer's Guide to Prime Networks* (DOC10113-1LA and UPD10113-11A)

● *Operator's Guide to Prime Networks* (DOC10114-1LA and UPD10114-11A)

● *User's Guide to Prime Network Services* (DOC10115-1LA)

● *NTS Planning and Configuration Guide* (DOC10159-1LA and UPD10159-11A)

● *NTS User's Guide* (DOC10117-1LA)

● *WSI300 User's Guide* (DOC10155-1LA and UPD10155-11A)

PRIME DOCUMENTATION CONVENTIONS

The following conventions are used throughout this document. The examples in the table illustrate the uses of these conventions.

Convention	Explanation	Example
UPPERCASE	In command formats, words in upper-case bold indicate the names of commands, options, statements, and keywords. Enter them in either uppercase or lowercase.	**SLIST**
italic	In command formats, words in lowercase bold italic indicate variables for which you must substitute a suitable value. In text and in messages, variables are in non-bold lowercase italic.	**LOGIN** *user-id* Supply a value for x between 1 and 10.
Brackets []	Brackets enclose a list of one or more optional items. Choose none, one, or several of these items.	**LD** $\begin{bmatrix} \textbf{-BRIEF} \\ \textbf{-SIZE} \end{bmatrix}$
Underscore in examples	In examples, user input is underscored but system prompts and output are not.	OK, <u>RESUME MY_PROG</u> This is the output of MY_PROG.CPL OK,
Ellipsis ...	An ellipsis indicates that you have the option of entering several items of the same kind on the command line.	**SHUTDN** *pdev-1* $\begin{bmatrix} ...pdev\text{-}n \end{bmatrix}$
Hyphen -	Wherever a hyphen appears as the first character of an option, it is a required part of that option.	**SPOOL -LIST**
Subscript	A subscript after a number indicates that the number is not in base 10. For example, a subscript 8 is used for octal numbers.	200_8

NEW FEATURES AT REV. 22.0

At Rev. 22.0 there is one new directive, MTRS, which sets the maximum tape record size for magnetic tapes. This directive allows the System Administrator to select the maximum record size available for reading and writing tapes via the T$MT programming call.

An operator command, CAB (for Change Async Buffers), replaces the AMLBUF, NTSABF, NTSBUF, and REMBUF directives for configuring communication lines. The CAB command allows dynamic reconfiguration and supports more processes and terminal lines.

A new process type, server, replaces phantom processes in several circumstances, with the result that fewer phantoms need be configured with the NPUSR directive.

OVERVIEW

This chapter describes:

- The responsibilities of the System Administrator

- Directories containing PRIMOS software

- Server processes and other system services

ROLE OF THE SYSTEM ADMINISTRATOR

A System Administrator organizes and manages computer systems. Therefore, a System Administrator must:

- Plan and set up each system, including the environments and attributes of the system's users

- Allocate system resources

- Set the policy for the use of the systems

- Make the systems secure

The System Administrator is the person to whom users and operators turn when anything goes wrong, or when problems arise unexpectedly. Although this book frequently refers to a single System Administrator, at your installation several people may share the job of administering the system. Alternatively, an operator or a user may double as the System Administrator.

The three volumes of the System Administrator's Guide cover many of the tasks that you, in your role as administrator, may be expected to perform. This volume shows you how to allocate disk space and software resources, including creation or revision of a system configuration file.

The *System Administrator's Guide, Volume II, Communications Lines and Controllers* explains the configuring of asynchronous lines, as well as I/O and buffer allocation.

The *System Administrator's Guide, Volume III, System Access and Security* explains how to:

- Determine the attributes of individual users and the user profiles they must have before they can log in to the system

- Set access rights on MFDs, system directories, and user directories to make the system secure

- Set up the Spooler and the Batch subsystems

- Set schedules for and perform backups

- Monitor system usage

OVERVIEW OF PRIMOS DIRECTORIES AND FILES

You should be familiar with the directories that comprise PRIMOS. Some directories and their associated files are delivered with all versions of PRIMOS. Others are delivered as separately chargeable products.

The next sections describe those directories that are nonchargeable and are required to run PRIMOS, those directories that are not required and are nonchargeable, and those directories that contain chargeable software and are not required to run PRIMOS.

The **command device** on a PRIMOS system is the disk partition having the logical device number zero. It is established by the COMDEV directive in the configuration file.

The Master File Directory (MFD) on the command device holds the top-level directories for all nonchargeable software, plus the following four important files.

Disk Record Availability Table
This file, sometimes called the DSKRAT, has as its filename the name of the partition on which it resides. It contains a table of the available records on the partition. This table is dynamic; that is, as disk records are used or freed, the table is constantly updated.

The MAKE command creates a new DSKRAT when making a partition. The PRIMOS file system uses the DSKRAT to keep track of available records, and the FIX__DISK command uses it in checking and repairing a partition's file structure.

BOOT
This file contains part of the bootstrapping procedure for cold starting the system.

BOOT__RUN__FILE__TREENAME
This file contains the pathname of the PRIMOS runfile used in the last successful boot of PRIMOS, in machine readable format. Do not attempt to edit this file.

BADSPT

A disk surface can have physical defects such as scratches or areas with little or no coating. The BADSPT file contains a list of all records that contain physical defects, or badspots. If a partition has no badspots, this file may be absent. When allocating disk records, PRIMOS scans the BADSPT file to ensure that no information is copied onto unusable records.

DYNBSP

Disks on intelligent controllers can be created with a dynamic badspot file, which holds a constantly-updated list of badspots.

Required Directories

The following top-level directories are required to run PRIMOS.

CMDNC0

The directory CMDNC0 contains external PRIMOS commands. External commands are those that are not internally embedded in the operating system; examples of external commands are ED and FIX_DISK. Frequently, this directory contains special commands that have been customized for your particular system. There are internal PRIMOS commands, such as ATTACH, RDY, and LOGOUT, that do not appear in CMDNC0.

CMDNC0 also contains the configuration file (typically named CONFIG) and the system startup file, normally named PRIMOS.COMI.

DOWN_LINE_LOAD*

DOWN_LINE_LOAD* contains the files that are loaded into the various controller subsystems when the system is booted.

DSM*

DSM* contains the files and directories needed for Distributed Systems Management. These are described in detail in the *DSM User's Guide*. The directory DSM*>LOGS is required for system and event logging to take place.

LIB

The directory LIB contains static-mode libraries.

LIBRARIES*

The directory LIBRARIES* contains library EPFs.

PRIRUN

The directory PRIRUN contains load maps and the PRIMOS runfiles, which are the files used to start up PRIMOS. This directory also contains the PRIMOS.COMI.TEMPLATE file.

SAD

The directory SAD (System Administration Directory) contains all user profile and project information. You can boot a system without a SAD, but users could not log in. You use EDIT_PROFILE to create the SAD, as described in *System Administrator's Guide, Volume III: System Access and Security*. This directory is created when you first invoke EDIT_PROFILE.

SEARCH_RULES*

The directory SEARCH_RULES*, created automatically when you run the installation program SYSTEM>ENTRY$.INSTALL.CPL, holds the default system entrypoint search rules file, ENTRY$.SR, and the ADMIN$.ENTRY$.SR file. It also holds several other search rules files, including ATTACH$.SR, BINARY$.SR, COMMAND$.SR, and INCLUDE$.SR.

SERVERS*

The directory SERVERS* contains the runfiles for the system servers such as the Login Server and the Auditor, if your system is using C2. This directory must be present.

SYSTEM

The directory SYSTEM contains all shared subsystem software, such as FORMS, compilers such as CBL, and utilities such as ED. SYSTEM contains the LOGIN_SERVER.COMI, which is run by the login server, and SET_LSR_ACLS.CPL, which sets ACLs on each directory in the System Administration Directory to grant LUR access to the Login Server. This directory also contains the command files INSTALL.STD.COMI, which installs PRIMOS, CREATE.ALL.COMI, which creates the directories required by chargable software, and INSTALL.ALL.COMI, which installs the chargeable software. Use these as templates, first deleting those products you will not be using, then running them to install your software.

UP_LINE_DUMP*

The LAN300 subdirectory of UP_LINE_DUMP* holds LHC dump files when they are generated.

Other Important Directories

The following directories may also be under your control, but are not required for running PRIMOS:

BACKUP*

Contains the files that comprise the BRMS utility, which is described in the *Data Backup and Recovery Guide.*

BATCHQ

Contains the files that are used whenever Batch jobs are run. These files include the Batch monitor runfile, Batch queue definition files, and job submittal files.

BOOTRUN

Contains the BOOT.INSTALL.COMI file, which installs BOOT from this directory into the MFD.

DIAG

Contains the files that comprise diagnostic tools for use by Customer Service Representatives.

DOS

Contains the obsolete single-user operating system, PRIMOS II, in the file DOS.SAVE.

INFO22.0

Contains the files that summarize the major changes in the current Revision. The name of this directory always matches the revision number.

HELP*

Contains HELP files for PRIMOS commands.

RJSPLQ*

Contains the files to run the Remote Job Entry (RJE) product.

SEG

Contains the files that build a SEG file to run as a command.

SEGRUN*

Contains segment directories (V-mode and I-mode runfiles).

SIT*

Contains the system internationalization tools for DSM.

SPOOL*

Contains ASCII files that control the environments of printer operations. This includes files that monitor the spooler and determine user privileges.

SPOOLQ

Contains the files L.DEST and L.TYPE on those systems running OAS. Otherwise, this directory is no longer used.

SPOOL_DATA*

Contains copies of the files to be printed (unless they were spooled with the -NOCOPY or the -SPOOL_WHILE_OPEN option). There can be several SPOOL_DATA* directories on a system, but only one per partition.

SPOOL_QUEUE*

Contains the list of print requests that are awaiting attention from a printer, and may contain a few optional files. There should be only one SPOOL_QUEUE* directory per system.

SYSCOM

Contains parameter insert files for compilers.

SYSOVL

Contains files required by CBL and the data files used by the FORTRAN 77, Pascal, PL/I, PL/I Subset G, and RPG compiler default driver programs. It also contains the EPF error table.

T&MRUN

Contains test and maintenance programs used by Customer Service Representatives.

TOOLS

Contains files and programs that can perform such tasks as converting your system to the current revision of PRIMOS. This directory also contains the driver programs for the PL/I Subset G, Pascal, and FORTRAN 77 compilers.

Some Optional Directories

Prime offers many additional products, most of which are stored in directories with names ending with the asterisk (*) symbol. Sample directories are:

FORMS*

Contains files needed to run the Forms Management System (FORMS). Must be installed to use FORMS. See the *FORMS Programmer's Guide.*

FTSQ*

Contains File Transfer Service (FTS) runfiles, the configuration data base, queues of transfer requests, and copies of user files for transfer. See the *PRIMENET Planning and Configuration Guide.*

NTS*

Contains the files necessary for Network Terminal Service.

PRIMENET*

Contains files needed to run PRIMENET networks, and the pre-Rev. 21.0 network event log files. See the *PRIMENET Planning and Configuration Guide.*

ACL Settings on System Directories

Most system directories require some access control to prevent unintentional or unauthorized use of their contents. See the *System Administrator's Guide, Volume III: System Access and Security* for a discussion of Access Control Lists and their use. Tables 1-1 and 1-2 provide recommended ACL values for system directories.

TABLE 1-1.   Access Rights for System Directories

Directory	Minimum Access Needed
BATCHQ	Protection is set by Batch subsystem. Must not be a password directory.
CMDNCO	$REST:LUR SYSTEM:ALL *System Administrator*:ALL recommended
DEVICE*	SYSTEM:ALL $REST:NONE
DSM*	.DSM$:ALL SYSTEM:LUR
DOS	SYSTEM:LUR
DOWN_LINE_LOAD*	LHC_DLL_SERVER:LUR LTS_DLL_SERVER:LUR SYSTEM:ALL *System Administrator*:ALL

Directory	Minimum Access Needed
HELP*	$REST:LUR
INF022.0	$REST:LUR recommended Directory name changes to match Rev.
LIB	$REST:LUR DALURW recommended for user modifying the libraries
LIBRARIES*	$REST:LUR DALURW recommended for user modifying the libraries
MFD (on command device)	.DSM$:LUR SYSTEM:ALL $REST:LU
PRIRUN	SYSTEM:LUR
SAD	Protection maintained by EDIT_PROFILE.
SEARCH_RULES*	$REST:LUR
SEGRUN*	$REST:LUR
SERVERS*	SYSTEM:LUR LOGIN_SERVER:LUR COPY_SERVER:LUR
SPOOL*	System Administrator:ALL .SPOOL_ADMINISTRATOR$:ALL $REST:LUR
SPOOL_DATA*	.SPOOL$$:ALL $REST:NONE
SPOOL_QUEUE*	.SPOOL$$:ALL .SPOOL_ADMINISTRATOR$:ALL $REST:NONE Must not be a password directory.
SPOOLQ	$REST:LUR
SYSCOM	$REST:LUR
SYSOVL	$REST:LUR
SYSTEM	SYSTEM:LUR $REST:LUR (for SYSTEM>DISCS)
UP_LINE_DUMP*>LAN300	LHC_ULD_SERVER:ALL LTS_DLL_SERVER:ALL

TABLE 1-2. *Access Rights for Special Products*

Product	Directory	Minimum Access Needed
DISCOVER	DISCOVER*	Should be a password directory
FED	FED*	$REST:RU *Installer*:ALL
FORMS	FORMS*	$REST:ALL
FTS	FTS	*System Administrator*:ALL
	FTSQ*	SYSTEM:ALL YTSMAN:ALL FTP:ALL RT_FTP:ALL *FTS Servers*:ALL $REST:DALURW
Networks	NETWORK_MGT*	$REST:LUR
NTS	NTS*	SYSTEM:ALL $REST:LUR
PRIMENET	PRIMENET*	NETMAN:UR RT_SERVER:UR *Network Administrator*:ALL SYSTEM:ALURWX $REST:NONE
	MFD containing PRIMENET*	NETMAN:U RT_SERVER:U
PRIME/SNA	PRIME/SNA*	See the *PRIME/SNA Administrator's Guide.*
PRIMIX	PRIMIX*	See *Using PRIMIX on the Prime 50 Series.*
RJE	RJSPLQ*	See the *Remote Job Entry Phase II Guide.*
ROAM	ROAM*	.ROAM_ADMIN:ALL SYSTEM:ALL $REST:LUR

SYSTEM SERVERS AND PHANTOMS

In normal operation, PRIMOS runs several server and phantom processes, each of which provides a service for all the users. Certain ones must be running for the system to operate, while others are optional, depending on the software you have installed. The servers shown in Table 1-3 must always be able to run.

For the system administrator, there is one important difference between phantoms and servers: you must configure, by using the NPUSR directive, the maximum number of phantoms allowed. On the other hand, no configuration of the number of servers is necessary, or even possible. Tables 1-3, 1-4, 1-5, and 1-6 show many servers and phantoms. You can always distinguish servers from phantoms by using the LIST_PROCESS -TYPE SERVER command.

Remember, when configuring phantoms, that you will need to allow additional phantoms for your users who may be using the PHANTOM or JOB commands.

The LOGOUT_SERVER process, new at Rev. 22.0, cleans up after logged-out processes, returning resources to PRIMOS for reuse. LOGOUT_SERVER itself cannot be logged out; it requires no startup and no configuration.

TABLE 1-3.   System Servers

Server	*Required By*	*Handles*
LOGIN_SERVER	PRIMOS	LOGIN and validation
LOGOUT_SERVER	PRIMOS	Cleanup after LOGOUT
DSMSR	DSM	Message Collection and request distribution
DSM_LOGGER	DSM	DSM logging process
SYSTEM_MANAGER	DSM	Event logging
TIMER_PROCESS	PRIMOS	Timed events

If your system is part of a network, you will also need some or all of the servers shown in Table 1-4. NM_SERVER is a phantom in spite of its name. All the others are servers.

TABLE 1-4.   Network Servers and Phantoms

Process	Required By	Handles
NETMAN	PRIMENET	PRIMENET support for Ring, Sync, PSDN, LAN300
ISC_NETWORK_SERVER	PRIMENET	Interserver Communications
NM_SERVER	NTS	Network Management
NTS_SERVER	NTS	LAN terminal server connection management

There are other phantoms and servers that may be running on your system depending on other software packages you may be using. Table 1-5 shows a few examples, all of which are phantoms, not servers.

Note

The names of the two FTS servers in Table 1-5 are the choice of the person who administers FTS. FTP and YTSMAN, shown here, are frequently used.

TABLE 1-5.   Additional Phantoms

Process	Required By	Handles
RT_SERVER	PRIMENET	Route through
FTP	FTS	File Transfer
YTSMAN	FTS	File Transfer
BATCH_SERVICE	BATCH	Batch jobs
Spooler phantoms	SPOOL	Printing of files

Additionally there are various temporary servers and phantoms started by the software to perform various tasks. Some of them are listed in Table 1-6. Only the COPY_SERVER is actually a server. The others, in spite of their names, are phantoms.

TABLE 1-6.   Temporary Servers and Phantoms

Process	Required By	Handles
COPY_SERVER	PRIMOS	Disk Mirroring
DSMASR	DSM	Applications running under DSM
LHC_DLL_SERVER	COMM_CONTROLLER	LHC downline load files
LHC_ULD_SERVER	COMM_CONTROLLER	LHC recovery
LTS_DLL_SERVER	COMM_CONTROLLER	Loading LTS300 controllers

DISKS AND TAPE DRIVES

Among the more important tasks of the System Administrator is the creation and allocation of disk space for system use and for users. Before your disks can be used for reading, writing, and updating information, the disks must conform to your system's requirements and your users' needs.

Here is what you have to know in order to set up your disk space for optimum efficiency and security:

- You must know the type and storage capacity of your disks.

- You must plan for dividing your total disk space into subdivisions (called partitions) and distributing the partitions on your system. Some partitions will hold files; others will be for paging.

- When you allocate paging space, you must make such decisions as the number of paging partitions to use, and the amount of space to allocate for paging.

- When you allocate user space you may set quotas (limits) on the number of records allocated to each top-level directory.

In addition to making decisions about disks, you must decide how to set up your magnetic tape drives. These disk and tape concerns are covered in this chapter.

Other related topics are covered elsewhere:

- The *Operator's Guide to File System Maintenance* describes how to format disks (with the MAKE utility) and how to repair the file system (with the FIX_DISK utility).

- The *System Administrator's Guide, Volume III: System Access and Security* describes how to monitor your system's disk space.

DISK TYPES AND STORAGE CAPACITIES

At Rev. 22.0, Prime supports two types of disk drives:

- Storage Module Disks (SMDs)

- Fixed-Media Disks (FMDs)

Each type is available in several storage capacities. For more information about all the disks that Prime supports, see the *Operator's Guide to File System Maintenance.*

Storage Module Disks (SMDs) are platters assembled into removable disk packs. The disk pack is inserted into and removed from its storage module drive. Prime supports two storage capacities for storage module disks: 80 and 300 megabytes, which have disk packs with 5 and 19 usable surfaces, respectively.

Fixed-Media Disks (FMDs), also called Winchester disks, are permanently enclosed dust-free drives. Prime supports several storage capacities for Winchesters: 60, 68, 84, 120, 158, 160, 258, 315, 496, 675, and 770 megabytes. The 68 and 158 megabyte versions are available only on the 2250™. The 60, 84, 120, and 258 megabyte versions are available only on the 2350™, 2450™ and 2455™ systems.

DIVIDING DISK PACKS

Before you format your disks, you must make three decisions:

- How to divide your total disk space into partitions

- The size of the partitions

- How to distribute the partitions on your disk controllers and disk drives

In making these decisions, you have two goals:

- To allocate space equitably among your users and allow for the system's needs for space (including reserving space for future expansion)

- To distribute the I/O workload evenly among your disk drives and controllers

Dividing Total Disk Area Into Partitions

To create partitions, you need to know the following information:

- The number of users in your various user groups. For example, how many users are in payroll, in manufacturing, or in inventory control?

- The nature of each group's work. How much storage space will each group require for its type of work?

- The workload of each group. Will the workload in each group be light or heavy six months or a year from now? How much storage space will each group require in the future?

- The software products that will be in use. Consult each product's documentation for suggestions on arranging disk storage.

- The amount of security and data reliability required by each group. For example, is there a reference database that must always remain unmodified, and is best kept on a read-only partition? Is there a constantly changing critical database that should be kept simultaneously on two separate disk drives, by use of the process called **mirroring**?

- The frequency of backups. For example, is there a small database requiring daily backups, and a large one for which weekly backups would suffice? Putting the two on separate partitions might substantially reduce the time required to perform daily backups.

- The number of disk drives and their storage capacities, as well as the number of controllers to handle the disk drives.

After you have collected this information and any other information that is important to your installation, you can decide how to partition your total disk space according to your users' needs.

Size of Partitions

Following are some guidelines for deciding whether to use large or small partitions.

Advantages of Using Large Partitions: Failing to grant enough disk space to a user partition at the time of the partition's creation is a common problem. Plan ahead when creating new user partitions, especially if your system is new. Allocate enough partition space so that you reduce the number of times the partition has to be moved, enlarged, or remade. It might be appropriate to set up all, or nearly all, of the space on each disk drive as a single partition.

Ideally, before creating the partitions, you should know which user groups are likely to have substantial increases in their workloads. You can thus allocate more space to those partitions. You can also use quotas to restrict space on large partitions. Other advantages of large partitions are:

- Holding large databases

- Being more efficient in storage

- Being more efficient in access time, due to reduced seek time

- Making it easier to reallocate space among directories

Advantages of Using Small Partitions: Smaller partitions provide the ability to write protect a database, by making a partition read-only.

Other advantages of small partitions are:

● Less data is lost if the partition is erased. That is, if most or all of the data on a small partition is somehow ruined or deleted, less data is lost than on a large partition.

● You gain flexibility in deciding how many directories to have online at any given time.

● You can isolate crucial data requiring mirroring or very frequent backup.

● Administration of short-term directories may be easier. For example the entire partition of student directories might be removed after every school term.

Some systems use small partitions to control the allocation of disk space among users. However, a more effective way of controlling the use of disk space is by setting quotas on top-level directories (as explained later in this chapter).

Backup Considerations: Disk-to-disk backups with COPY_DISK, require source and target partitions to be of equal size. Therefore, you might want to standardize the sizes of your partitions as much as possible.

For more information on backups, see the *Data Backup and Recovery Guide* and the *System Administrator's Guide, Volume III: System Access and Security.*

Distributing Partitions to Drives and Controllers

When adding new partitions to your system or adjusting existing partitions, follow this rule of thumb: Distribute the use of your partitions evenly among your disk drives and distribute your drives evenly among your controllers. An even distribution makes read/write operations faster and the system more efficient. In particular, if two (or more) partitions are often accessed simultaneously, placing them on separate controllers may greatly improve performance. This rule applies equally to file storage and to paging. See page 2-6 for further guidelines.

For example, if you have five partitions, two drives, and two controllers, you might place the three smaller partitions on one drive and the two larger partitions on the other, thereby balancing the data distribution as much as possible. Then, you would place one drive on each controller, so that read/write operations on both drives could occur simultaneously.

If you subsequently discover that read and write activity to the smaller partitions is far more frequent than to the larger ones, you might want to divide the smaller partitions among the two drives, thereby balancing the I/O activity as much as possible. Each drive would then hold one large partition and one or two smaller ones, or one lightly used partition and one or two heavily used partitions.

Monitoring the Distribution: Keeping your partitions and drives evenly distributed is an ongoing process and requires that you do the following:

● Monitor the data distribution regularly.

● Watch the trends and patterns in the way your users manipulate their storage space. For example, if one partition's workload increases, more data is added to the partition and the read/write operations on that partition increase substantially.

● Be prepared to adjust the data distribution so that the increase in read/write operations does not hamper system efficiency.

Five PRIMOS commands that monitor system operations and data storage information are AVAIL, LIST_QUOTA, PRIMON, STATUS, and USAGE. These commands, along with other commands and information on monitoring the system, are discussed in the *Operator's Guide to System Monitoring*, and in the *System Administrator's Guide, Volume III: System Access and Security*.

Two of the SIM commands, described in the *DSM User's Guide*, are particularly helpful for monitoring disk and file usage. LIST_DISKS displays the status of the disk partitions, and LIST_UNITS displays the names of each user's active files.

Pre-Rev. 22.0 Partitions

Although you can use Rev. 18, Rev. 19, Rev. 20, and Rev. 21 partitions on a Rev. 22.0 system, to gain the advantages of the file structure introduced at Rev. 21.0, you should convert pre-Rev. 21.0 partitions to Rev. 22.0 format. Hashed directories, and the Date/Time Created and Date/Time Accessed attributes for files and directories were introduced at Rev. 20.0, quotas and ACLs at Rev. 19.2.

To convert pre-Rev. 21.0 partitions to Rev. 22.0-format partitions, use the MAKE utility. Also, although it is not likely that you will need to change Rev. 21.0-format partitions to Rev. 18-format or Rev. 19-format, you can perform such a conversion with the -DISK_REVISION option of MAKE. Before attempting any conversions, read the *Operator's Guide to File System Maintenance* carefully.

A Rev. 22.0, 21.0 or 20 partition cannot be added locally on a pre-Rev. 20 system, but it can be added remotely. Users logged in on a pre-Rev. 20 system and attached to a Rev. 20 or later partition cannot display the Rev. 20 Date/Time Created and Date/Time Accessed file and directory attributes with the LD command.

ALLOCATING PAGING SPACE

As System Administrator, you must make sure that your system has enough paging space, not only when you first configure it, but later, when you have added more users. If there is not enough paging space, you may find that users cannot log in when the system is being heavily used.

To do this, you must make two decisions: how much paging space to allocate, and the number of paging partitions to use. To create the partitions for paging, use the MAKE utility, described in the *Operator's Guide to File System Maintenance*. For additional information concerning conversion of pre-Rev. 20 paging partitions, see the handbook for your CPU model.

An Overview of Paging

Here is a simplified explanation of paging. Programs execute in a computer's main, high speed memory. Because the amount of this memory is limited, the PRIMOS operating system uses **paging space** (disk space on **paging partitions**) as secondary memory. The main memory, the secondary memory, and the programs are divided into **pages** of 2048 bytes. For a program to execute, only those pages of it containing the current instruction and the data used by that instruction need be in main memory. When an instruction refers to data that is not in main memory (or when the next instruction is not in main memory), the appropriate pages are brought in as needed. This method is called **demand paging**. To make space for the pages being brought in, other pages, less recently used, may need to be written out to the paging space.

EPFs and Paging

EPFs have filenames ending in .RUN; you can find EPF programs in CMDNC0, and libraries for EPF programs in LIBRARIES*. When paging occurs for an EPF program, pages of per-user data are read from (or written to) the paging partitions as required. The pages of sharable data (such as executable code) are never written out at all, because they never change. Instead, they are read directly from the .RUN files as required.

Paging of an EPF program often requires simultaneous disk activity both on the COMDEV (the partition where CMDNC0 and LIBRARIES* reside) and on the paging partitions. If there is a paging partition on the same physical disk that holds the COMDEV, the heads of that disk may spend much time seeking between the COMDEV and paging partitions. It is usually a good idea to have paging partitions and the COMDEV on separate disks, or even separate disk controllers.

The executing user program cannot detect that it is being paged. It sees no distinction between main memory and secondary memory; indeed, it has available to it an addressing space much larger than the main memory. This feature, along with the memory management scheme for implementing it, is sometimes called **virtual memory**.

For a more accurate, and much more detailed, explanation of memory management in Prime computers, see the *System Architecture Reference Guide*.

Running Out of Virtual Memory

A system can run out of virtual memory for one of three reasons. Each requires you to take a different action.

- If a user runs out of segments (indicated by a "Not enough segments" message), you may need to use EDIT_PROFILE to increase the number of static or dynamic segments for that user.

- If a system runs out of system segments (indicated by the user error condition NO_AVAILABLE_SEGS$), you may need to increase the NSEG or NVMFS configuration directives.

- If the paging partition becomes full (indicated by the user condition PAGING_DEVICE_FULL$), you must add more paging partitions or increase their size.

Establishing Paging Partitions

A system can use up to eight partitions for paging. Use the PAGING directive, discussed in Chapter 6, Configuration Directives, to tell PRIMOS which partitions are for paging.

Notes

The PAGING directive replaces the now-obsolete PAGDEV and ALTDEV directives. See Appendix A, Obsolete and Rarely Used Commands and Directives.

Also, the PRATIO operator command has replaced the obsolete PRATIO directive. See the *Operator's Guide to System Commands* for more information.

The paging partitions should, if possible, be on disk drives that are not used heavily for other purposes. For instance, there should not be a paging partition on the drive that holds the command device partition (the COMDEV). Multiple paging partitions may help divide the loads among the disk drives and controllers. There is, of course, no advantage in putting more than one paging partition on any disk drive. For help in making these decisions, consult your Prime Customer Service Representative.

Paging Space Requirements

Paging space is allocated in units of 16 kilobytes. This means that when the first eight pages of a segment are accessed, only 16 kilobytes of paging space are used by the segment. Therefore, a given amount of paging space can accommodate a varying number of segments, depending on the number of pages used in each.

Because PRIMOS cannot determine whether the amount of paging space is adequate for the number of available system segments (set by the NSEG configuration directive), paging space may be exhausted while the system is running. If paging space is exhausted, the user requesting the additional memory receives the error condition PAGING_DEVICE_FULL$.

Determining the Amount of Paging Space

There are two methods for determining the amount of paging space:

- Use the rule of thumb, below.

- Calculate the maximum and minimum amounts of paging space your system could require, using the formulas given in the sections that follow. Your optimal paging space will fall somewhere between the two. For small systems and lightly loaded systems the paging space should probably be closer to the minimum figure, and for large or heavily loaded systems, closer to the maximum.

Rule of Thumb: A good rule of thumb for determining the amount of space you need for paging is to allocate 7800 records (or 16 megabytes, that is, one disk surface of a 300 megabyte disk) for paging for every 6 to 8 users. The number of users is the sum of the NTUSR, NRUSR, NTSUSR, NPUSR, and NSLUSR configuration directives, plus the number of server processes. See page 1-9 for information on server processes.

Calculating Maximum Paging Space: The formula for calculating the maximum amount of paging space needed on your system is:

$$MAX\_SPACE = NSEG * 64$$

where

MAX_SPACE
   is the maximum paging space needed (in records).

NSEG
   is the total virtual address space for the system, as set by the NSEG configuration directive.

64
   is the number of pages per segment.

Calculating Minimum Paging Space: The formula for calculating the minimum amount of paging space needed is:

$$MIN\_SPACE = PRIMOS + SHARED\_PRODUCTS + (NUSR * 304)$$

where

MIN_SPACE
   is the minimum amount of paging space (in records) your system requires.

PRIMOS
   is the number of pages used by PRIMOS. This is 1552 at Rev. 22.0.

SHARED_PRODUCTS
   is the total number of pages used by the shared products on your system. Table 2-1 lists the number of pages per product and pages per user needed for each shared product. Add the per-product figures for each shared product in use on your system. For each product, multiply the per-user pages by the expected number of simultaneous users for

that product (not by the total number of users on the system). Use the total of these figures as the number for SHARED_PRODUCTS.

NUSR
   is the sum of the configuration directives NTUSR + NPUSR + NRUSR + NTSUSR + NSLUSR, plus the number of servers.

304
   is 38 segments per user * 8 pages per segment.

TABLE 2-1.   Space Required by Shared Products

Product	Per-Product Pages	Per-User Pages
BASIC	0	48
BASICV	56	24
CBL	376	184
DBG	224	64
DBMS	194	178
DPTX	53	(see Notes)
ED	24	8
EDB	0	8
EMACS	448	52
FED/FORMS	160	280
FTN	0	40
FTS	152	272
LOAD	0	16
MIDASPLUS	320	200
PMA	0	16
PSD	0	8
RJE	0	(see Notes)
ROAM	288	48
RPG	0	32
RUNOFF	0	48
SEG	0	40
SORT	0	24
VPSD	0	16

Notes for Table 2-1.

DPTX   The per-system value assumes a maximum configuration of 7 emulators running and 1 line for support use. The per-user value depends on the type of terminal in use. Values are: for the PT45, 64; for the PT46, 56; for the OWL, 53; for the PST 100 and PT200, 60.

RJE   To calculate per-user paging space for RJE, allow 208 pages for the common runfiles, plus 168 pages for each emulator you use (1004, 200UT, 7020, GRTS, HASP, X80, XBM).

Split Paging Disks

A paging partition is called a **split disk**. It is split between paging space and file system storage, so that there is room for badspot files. See the *Operator's Guide to File System Maintenance* for several important instructions on constructing paging partitions, including the use of the MAKE command.

ALLOCATING USER SPACE WITH QUOTAS

Ensuring equitable sharing of disk storage among users is a primary function of the System Administrator. You can provide that equity by setting limits (called **quotas**) on the amount of storage space that directories occupy on a partition.

The quotas, which are measured and allocated by the number of disk records, can be set by both the System Administrator and the user with the SET_QUOTA command. As the System Administrator, you are responsible for setting and modifying the quotas on top-level directories.

Note
A quota cannot be placed on an MFD.

After you have set quotas on your system's top-level directories, users can set or modify quotas on subdirectories only if they have Protect rights (in ACL directories) or Owner rights (in passworded directories) to the next higher directory. That is, the user must have the appropriate rights to the directory that contains the subdirectory whose quota is to be set.

Users can find instructions and guidelines for setting and modifying quotas in the *PRIMOS User's Guide.*

Four Strategies for Setting Quotas

The amount of disk space on a partition that is reserved for users is the number of records remaining after you allocate space to paging and to mandatory PRIMOS files and directories. After you have determined this space, you can use one of four strategies discussed below to distribute and manipulate user disk space. The strategies all include setting quotas on top-level user directories.

Set quotas on top-level directories according to how structured you want your user space to be. That is, decide whether to set strict limits on each user user (or user group), or whether to set looser limits within which users compete for the disk space.

You can use any of four major strategies for setting quotas on top-level directories:

- The **Exact** strategy, in which the sum of the quotas on the top-level directories equals the capacity, of the partition.

- The **Undercommitted** strategy, in which the sum of the top-level quotas is less than the capacity of the partition.

- The **Overcommitted** strategy, in which the sum of the top-level quotas is greater than the capacity of the partition.

- The **Unregulated** strategy, in which one or more directories on the partition has no quota.

The Exact Strategy: The Exact   strategy distributes all the disk's space precisely among users.   Each user is guaranteed his or her entire quota.   No user's usage of disk space affects other users.   No user encounters "Disk Full" errors, because the quota limit is always reached first.

For example, suppose your partition (MFD) has a capacity of 100,000 records that are reserved for users' work space, and that you plan to install four directories, as shown in Figure 2-1. Taking a strict approach, you could ensure that your users never use up more than 100,000 records by setting quotas that total the capacity of the partition. You might give one directory 40,000 records and the other three directories 20,000 records each. After setting the quotas, you would monitor which top-level directories were using their space and modify the quotas accordingly.

The Undercommitted Strategy: The Undercommitted strategy is as strict as the Exact strategy, providing the same guarantee of available disk space to the users. Additionally, it keeps space in reserve for expansion or emergency use.

The Exact and Undercommitted strategies both create an incentive for the users to be more efficient, reserving their space for essential data and deleting unneeded data.

Using the 100,000-record partition of the previous example, you could set a quota of 15,000 records on each of the four directories, thus ensuring that you would always have 40,000 records in reserve.

The Overcommitted Strategy: The Overcommitted strategy   is less rigid than the Exact or the Undercommitted. This strategy, in which the sum of the directory quotas is greater than the capacity of the disk, causes users to compete for disk space. The Overcommitted strategy may result in more complete utilization of the partition's space than does the Exact strategy.

Some disadvantages to the Overcommitted strategy are:

- The partition can become completely filled, resulting in "Disk Full" errors, before any users have exceeded their quotas.

- Users may waste space by keeping unnecessary files around, because they perceive their usage to be far below the available quota.

- A quota does not guarantee space to a directory.   Instead, a directory is guaranteed space only to the extent that size of the partition exceeds the sum of the quotas for the *other* directories.   Here is how guaranteed space works out if we apply quotas to the partition in Figure 2-1 using the Overcommitted strategy.   Remember, although the
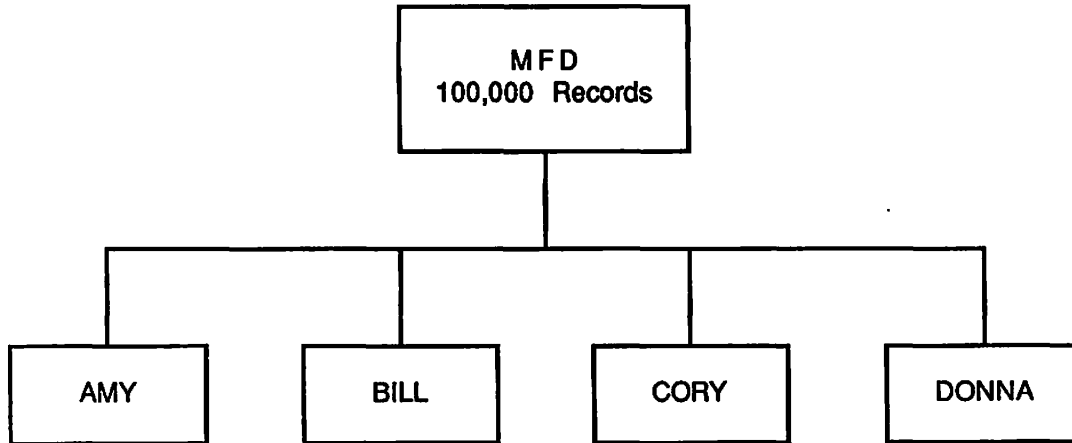
FIGURE 2-1. Directory Structure for Disk Quota Examples

sum of all the quotas in this example is 145,000 records, the size of the partition is only 100,000.

Directory:	AMY	BILL	CORY	DONNA
Quota:	20,000	65,000	10,000	50,000
Sum of other quotas:	125,000	80,000	135,000	95,000
Guaranteed space:	none	20,000	none	5,000

The Unregulated Strategy: The least rigid strategy is the Unregulated strategy, where one directory (or more) has no quota at all, or has a quota larger than the size of the partition. A directory has no quota in either of two circumstances:

● The directory has never been assigned a quota.

● The quota size has been set to zero.

You can improve system performance when using the Unregulated strategy by setting an extremely large quota instead of no quota on each unregulated directory. The large quota should exceed the partition size. See the section below, Speeding Up the LD and LIST_QUOTA Commands.

The storage capacity of a nonquota directory is limited only by the physical capacity of the partition. Setting no quotas on directories gives users the impression that their allotment of disk space is unlimited. If any directory on a partition has no quota, then no other directory on that partition has guaranteed space.

You might use the Unregulated strategy if you have a special user group whose members, by the nature of their work, must be trusted with a seemingly unlimited amount of disk space. With a 100,000-record partition, you might set two of your directories to 20,000 records each, and give the two others no quota. None of the four directories would have any guaranteed amount of disk space.

Speeding Up the LD and LIST_QUOTA Commands

You can improve the performance of the LIST_QUOTA and LD commands significantly by placing a quota on top-level directories. A quota causes PRIMOS to maintain up-to-the-minute quota information at no performance cost. Quota information is therefore readily available and does not have to be collected each time a user issues the LIST_QUOTA and LD commands. Performance is particularly improved for very large directory structures.

To improve performance without restricting space, use a very high quota (such as 1,000,000), which in essence removes any quota restriction on the directory.

Monitoring Quotas

After you set quotas on top-level directories, you should monitor the directories to determine how many records are being stored in them. If necessary, you may have to modify the quotas. To monitor the use of space in directories, use the LIST_QUOTA, LD, and SIZE commands.

The LIST_QUOTA command lists the maximum quota on a directory, the total number of records used by the entire subtree (beginning with and including the designated directory), and the number of records used by this particular directory. For details on LIST_QUOTA, see the *PRIMOS User's Guide* and the *System Administrator's Guide, Volume III: System Access and Security*.

The LD command also supplies information on quotas and record usage. The SIZE command lists the size of directories and files. For more information on these commands, see the *PRIMOS User's Guide* and the *PRIMOS Commands Reference Guide*.

To modify a quota, use the SET_QUOTA command.

Calculating Storage Availability: To determine how much storage space is left in a directory, you must consider all quotas set on the entire directory tree and also the total current storage used by the entire directory tree.

See the *PRIMOS User's Guide* for explanations and illustrations of how to calculate storage availability.

Recovering From Quota Overloads: If a user tries to store data that will cause a quota to be exceeded, PRIMOS returns the message `Maximum quota exceeded` and does not allow storing the material.

For information on how to recover from quota overloads (including those that occur during an editing session), see the *PRIMOS User's Guide*.

MAGNETIC TAPE DRIVES

The SETMOD command controls assignment of tape drives.   Use this command to allow users to assign tape drives from their terminals, to require them to ask the operator, or to prevent use of tape drives.

The SETMOD command has three formats:

- SETMOD -USER (which is the default state for the system) allows users to perform their own tape operations.   Users can issue the ASSIGN command either to assign tape drives to themselves, or to request the operator to perform the tape operation (which includes assigning the tape drive and setting its characteristics).   The latter choice allows phantom jobs and batch jobs to run under operator control, while interactive jobs can run under operator or user control.   Either the user or the operator can use the UNASSIGN command to unassign a tape drive that a user has assigned.

- SETMOD -OPERATOR causes the ASSIGN command to channel all requests for tape drives to the supervisor terminal.   The operator must approve or disapprove each request.   Either user or operator can UNASSIGN a tape drive after it is assigned to a user.   This mode is appropriate if you do not permit users to enter the computer room.

  To set up your system to function in this mode as a matter of course, add the SETMOD -OPERATOR command to your PRIMOS.COMI file, so that the command is invoked when the system is cold started.

- SETMOD -NOASSIGN prohibits all tape drive assignments.   When the system is in -NOASSIGN mode, an attempt to ASSIGN a tape drive produces a message stating that tape drives cannot be assigned at the present time.   To make tape drives available again, use the SETMOD command with either the -USER or -OPERATOR option.

  Use the -NOASSIGN mode when the operator is not available to handle tape requests or when you want no tape operations conducted.

If you use -USER mode, you must still decide whether to allow users to enter the computer room to load and unload tapes (and perhaps to keep them in your tape storage facility), or whether to allow only operators to perform these tasks.

See the *Operator's Guide to System Commands* for the complete instructions on using the SETMOD command, and the *Operator's System Overview* for examples.

INITIALIZING SYSTEM SOFTWARE

This chapter contains information on ensuring the proper initialization of shared system software:

- Shared segments, including a table of the segments to which Prime has assigned products, the segments reserved for Prime, and the segments specifically reserved for customer use

- Library EPFs, including a description of the system entrypoint search list

- Shared static-mode libraries, including a table of the shared static-mode library package numbers

Programmers and analysts who need to know the details of shared segments, EPFs, and library EPFs should read the *Advanced Programmer's Guide, Volume 1: BIND and EPFs.*

SHARED SEGMENTS

Shared subsystems normally are initialized at cold start by the PRIMOS.COMI file. After cold start you can initialize individual shared subsystems supplied by Prime by using this command at the supervisor terminal:

   CO SYSTEM>*name*.SHARE.COMI

Software supplied by other vendors may use the segments reserved for customers. See Table 3-1. As System Administrator, you are responsible for assigning and coordinating the use of these segments. For shared subsystems not supplied by Prime, use this command from the supervisor terminal:

   SHARE *pathname segment-number [access-rights]*

where *pathname* is the pathname of the file whose contents are to be loaded into segment *segment-number*, and *segment-number* is the octal number of the segment to be shared. See Table 3-1 for a list of segments specifically reserved for customer-shared subsystems.

access-rights is a number that specifies user access to the segment. The valid values for *access-rights* are as follows:

     0   No access

    200   Read access only

    600   Read and execute access (default)

    700   Read, write, and execute access

See the *Operator's Guide to System Commands* for details on the SHARE command.

Caution

If you use the SHARE command incorrectly, the result may be that user programs can overwrite the operating system and the shared utilities. Do not share into segments 0 - 1777, which are reserved for PRIMOS. Other segments that may contain system utilities are listed in Tables 3-1 through 3-4.

TABLE 3-1.   Contents of Shared Segments at Rev. 22.0

Segment	Product
2000	ED
2001-2003	DBMS
2004-2011	SPSS
2012	DBMS
2013	BASIC/VM
2014	Reserved for Prime
2015	DPTX
2016	Reserved for Prime
2017	BASIC/VM
2020	Reserved for Prime
2021	FORMS library
2022-2023	Reserved for Prime
2024-2025	PRIME/POWERPLUS
2026-2027	FTS
2030-2037	**Reserved for customers**
2040-2042	DBG
2043	SPSS
2044-2056	Reserved for Prime
2057-2065	OAS
2067	Reserved for Prime
2070	DBMS
2071	OAS
2072	SPSS
2073-2077	DISCOVER™
2100	EDMS™
2101	OAS
2102-2114	EDMS
2115	DBG
2116-2121	Reserved for Prime
2122-2125	MIDASPLUS™
2126-2127	FTS
2130-2137	PRIME MEDUSA™
2140	EDMS, BP99
2141-2150	Reserved for Prime
2151-2153	FED
2154-2161	CBL
2162-2163	EDMS, BP99
2164-2166	Reserved for Prime
2167	SPOOL
2170-2177	**Reserved for customers**
2200-2203	ROAM
2204-2207	PRISAM™
2210-2215	ESCAPE34
2216	Reserved for Prime
2217-2220	ROAM
2221	Reserved for Prime
2223-2224	ROAM
2225	Reserved for Prime
2226	ESCAPE34

Segment	Product
2227	PRISAM
2230-2267	PRIMEWAY™
2270-2276	Prime INFORMATION™
2277	DISCOVER
2300-2317	**Reserved for customers**
2320-2321	MIDASPLUS
2322	Reserved for Prime
2323	PRIMEWAY
2324-2327	C
2330-2337	Prime INFORMATION
2340-2347	EMACS
2350-2367	PDGS
2370-2376	PRIME MEDUSA
2377	PRIME/SNA
2400-2427	PDMS™
2430-2442	THEMIS
2443	EDMS
2444-2447	Reserved for Prime
2450-2467	PRIMEWAY
2470-2475	Prime INFORMATION CONNECTION
2476	PRIME/SNA RJE
2477	Reserved for Prime
2500-2521	Prime ORACLE
2522-2534	Reserved for Prime
2535	CBL
2536-2547	Reserved for Prime
2550-2556	C
2557-2564	PDGS
2565-2567	Reserved for Prime
2570-2573	ESCAPE
2574-2575	Reserved for Prime
2576	DBG
2577	Reserved for Prime
2600-2601	ROAM/DDM
2602-2665	Reserved for Prime
2666-2765	Reserved for EPFs
6001 } 6006 } 6007 }	(Per-user linkage segments. See tables 3-2, 3-3, 3-4.)
6010	ORACLE, EMACS, PRIMEWAY
6011	ROAM
6012	Reserved for Prime
6013	INFORMATION
6014	Reserved for Prime
6015	Reserved for Prime
6016	(See table 3-5.)
6017	ORACLE

TABLE 3-2.   Segment 6001

Allocated	Product
0-32777	FORMS
33000-66777	Reserved for Prime
67000-67767	SPOOL
67770-67777	BATCH
70000-105777	FORMS
106000-112777	ED
113000-117777	NPX
120000-131777	ABBREV
132000-177777	FORMS

TABLE 3-3.   Segment 6006

Allocated	Product
0-37777	FTS
40000-70000	MIDASPLUS
70001-77777	Reserved for Prime
100000-177777	ROAM/DDM

TABLE 3-4.   Segment 6007

Allocated	Product
0-47777	ROAM
50000-122777	PRISAM
123000-137777	Reserved for Prime
140000-177777	MAGLIB

TABLE 3-5.   Segment 6016

Allocated	Product
0-37777	Reserved for Prime
40000-177777	ORACLE

LIBRARY EPFS

A library EPF is a set of subroutines that are bound together (with the BIND linker) into one file. Subroutines within the file that are entrypoints are available to the PRIMOS dynamic linking mechanism.

Features of Library EPFs

Library EPFs share the following advantages with shared static-mode libraries:

● User runfiles are smaller, thus reducing the time required for invocation. User interaction with the program begins sooner.

● System load is reduced with respect to private segments and private memory image sizes, and paging may also be reduced. System load reduction is important for large V-mode and I-mode programs that make extensive use of system library routines.

● Installation of a new revision of the library does not require program reloading. Installation of a rebuilt library is all that is required to make the modified library available to all users of the library.

In addition, library EPFs provide the following advantages over shared static-mode libraries:

- Library EPFs are not shared into static segments, but instead are brought into memory automatically when the dynamic linking mechanism must link to an entrypoint in the library. Therefore, the System Administrator does not have to coordinate the use of shared segments among library EPFs.

- Library EPFs do not require that the system be shut down and restarted to install a new version of the program reliably. (This procedure is recommended when installing a shared static-mode library because unrecoverable errors usually result if a user is executing the old version of the program when the new version is installed.)

- Library EPFs are not loaded into the segments at cold start.

- Users can create their own library EPFs and use ACLs to restrict their use as desired.

For more information on library EPFs, see the *Programmer's Guide to BIND and EPFs* and the *Advanced Programmer's Guide, Volume 1: BIND and EPFs.*

Installation of Supplied Library EPFs

The directory LIBRARIES* contains the library EPFs supplied by Prime. Unlike static-mode libraries, these library EPFs do not need to be shared with the SHARE command.

To install a new library EPF in the LIBRARIES* directory, use the following procedure:

1. Use the COPY command to copy the EPF into LIBRARIES*.

2. Use a text editor (such as ED or EMACS) and add the name of the library to the system entrypoint search list. (The entrypoint search list, named ENTRY$.SR in the SEARCH_RULES* directory, is described in the next section.)

To replace an existing library EPF in the LIBRARIES* directory, use the COPY command. See the section about Adding EPF Programs on page 4-2 for details and examples.

When you are replacing an existing library EPF, some users may be using it; that is, the existing library may be mapped into those users' address spaces. In this case, those users continue to use the existing library, and are unaware of the change. (Users invoking the library after the change get the new library.) Moreover, the existing library is not deleted, but becomes a **replace file**. Its filename suffix is changed from .RUN to .RPn, where n is a digit from 0 to 9.

If libraries are replaced frequently, you should periodically clean up the LIBRARIES* directory by deleting the old replace files (those files with the .RPn suffix) that are no longer in use.

SYSTEM ENTRYPOINT SEARCH LIST

The system entrypoint search list determines the order in which PRIMOS searches libraries to find a match to a subroutine entrypoint in a program. The template file for the default entrypoint search list, ENTRY$.SR, is kept in the directory named SEARCH_RULES*. The installation program SYSTEM>ENTRY$.INSTALL.CPL automatically creates the SEARCH_RULES* directory, if it does not already exist, and copies SYSTEM>ENTRY$.SR into it. You can delete the copy of ENTRY$.SR file in the SYSTEM directory after you check that ENTRY$.SR is in the SEARCH_RULES* directory.

In addition to containing ENTRY$.SR, the SEARCH_RULES* directory must also contain a file called ADMIN$.ENTRY$.SR. This file, which the installation program also puts in place, contains the following single rule:

    -PRIMOS_DIRECT_ENTRIES

You should never modify the ADMIN$.ENTRY$.SR file.

Format of the Search List

The system entrypoint search list template file is a text file that contains a list of search rules (one search rule per line). A search rule has one of of the following two formats:

- The pathname of a systemwide library EPF. For example:

    LIBRARIES*>FTN_LIBRARY.RUN

- A keyword that begins with a hyphen. For example:

    -STATIC_MODE_LIBRARIES

Because SEARCH_RULES*>ENTRY$.SR is a text file, you can modify it with a text editor or display its contents with the SLIST command, as in this example. (Your file may differ from the one shown here.)

```
OK, SLIST SEARCH_RULES*>ENTRY$.SR
/* ENTRY$.SR, SYSTEM, JNS, 11/04/87
/* ENTRY Search list
/* Copyright (c) 1985, Prime Computer, Inc., Natick, MA 01760
LIBRARIES*>TRANS_LIB$PRC.RUN
LIBRARIES*>TRANS_LIB$PRG.RUN
LIBRARIES*>SYSTEM_LIB$PRG.RUN
LIBRARIES*>SYSTEM_LIB$PRC.RUN
LIBRARIES*>APPLICATION_LIBRARY.RUN
LIBRARIES*>COMMON_ENVELOPE.RUN
LIBRARIES*>PRIMOS_LIBRARY.RUN
LIBRARIES*>MATRIX_LIBRARY.RUN
LIBRARIES*>ECL$LIB.RUN
LIBRARIES*>CURSLB.RUN
LIBRARIES*>FTN_LIBRARY.RUN
```

```
LIBRARIES*>MAGTAPE_LIBRARY.RUN
LIBRARIES*>VT$.RUN
LIBRARIES*>SORT_LIBRARY.RUN
-STATIC_MODE_LIBRARIES
LIBRARIES*>CC_LIBRARY.RUN
LIBRARIES*>CBL_LIBRARY.RUN
LIBRARIES*>PASCAL_LIBRARY.RUN
LIBRARIES*>PL1G_LIBRARY.RUN
LIBRARIES*>VRPG_LIBRARY.RUN
LIBRARIES*>DSMLIB.RUN
LIBRARIES*>DSMSITLIB.RUN
LIBRARIES*>SIT_LIBRARY.RUN
LIBRARIES*>X409LIB.RUN
LIBRARIES*>SP$LIB.RUN
LIBRARIES*>DRBLIB.RUN
LIBRARIES*>DRB_PROCLIB.RUN
LIBRARIES*>LISP_LM.RUN
LIBRARIES*>CC_LM.RUN
LIBRARIES*>PL1_LIBRARY.RUN
LIBRARIES*>PL1_SYSTEM_LIBRARY.RUN
libraries*>start_stop_nmsr.run
```

Search Order

The order in which the search rules are listed in the ENTRY$ search list is the order in which PRIMOS searches the libraries to find a match to a subroutine entrypoint. Typically, the order indicates that systemwide library EPFs (in the LIBRARIES* directory) are to be searched first (after internal PRIMOS entrypoints, which are always searched before any libraries listed in the entrypoint search list).

At some point, the search list usually contains the search rule -STATIC_MODE_LIBRARIES, which directs that the static-mode libraries are to be searched. Although Prime supplies several individual static-mode libraries, these libraries are treated as one entity.

Because the order of the search rules determines the order in which the libraries are searched, a proper ordering improves the speed at which the subroutine is found. The library for a frequently called subroutine should be listed so that it requires the shortest search time possible.

The search order is also important when naming conflicts occur between libraries. The order in which the conflicting entrypoints appear determines which copy of a subroutine is actually invoked. It is best, of course, to avoid such naming conflicts altogether.

Access Rights on the Search list

Set access rights on the system entrypoint search list SEARCH_RULES*>ENTRY$.SR so that only you (or someone designated by you) can modify the file. You might use the following access rights:

```
SYSTEM: ALL
$REST:LUR
```

User Entrypoint Search Lists

Users can create their own entrypoint search lists and enable them with the SET_SEARCH_RULES command. A user's entrypoint search list automatically includes SEARCH_RULES*>ENTRY$.SR unless the user's command line contains the -NO_SYSTEM option and the user's list does not include -SYSTEM.

Users can display their current entrypoint search list by using the LIST_SEARCH_RULES command (abbreviated LSR).

If there are a number of libraries, each of which is used by only a few people, it may be better for those people to have their own entrypoint search lists, rather than for those libraries to be in the default search list. The latter situation would require all users to have enough dynamic segments to map in everything on the entrypoint search list, or risk getting the error condition "Not enough segments." (The EDIT_PROFILE command sets the number of dynamic segments per user.) A short entrypoint search list usually results in better performance.

Caution

You should usually encourage users to use the system copy of ENTRY$.SR (which is obtained automatically) rather than maintaining and using a private copy. If a user needs a private copy, the user should do one of the following:

- To include system rules at the beginning of the list, use the SET_SEARCH_RULES command without the -NO_SYSTEM option.

- To put the system rules other than at the beginning, put the keyword -SYSTEM in the list if the list does not contain any of the rules in the system copy of ENTRY$.SR.

Some users may desire to have search rules that are unrelated to the system copy of ENTRY$.SR. (They would do this by using the -NO_SYSTEM option to SET_SEARCH_RULES, and by omitting the -SYSTEM rule from their ENTRY$.SR files.) You should make them aware that it is their own responsibility to keep their ENTRY$.SR files up to date if the system copy is changed.

Linkage Faults

If the end of the search list is reached without the target subroutine having been found, or if the ENTRY$ list has been altered or improperly installed, the dynamic linking mechanism signals the condition LINKAGE_FAULT$. The linkage fault normally produces an error message such as the following:

```
Error: condition "LINKAGE_FAULT$" raised at 4243(3)/1031.
Entry name "GET_LINE" not found while attempting to resolve
dynamic link from procedure "FIND_NUM".
ER!
```

The following steps should remedy the condition:

1. The user should enter the following command to reinitialize the search rules to system default:

   ```
   SET_SEARCH_RULES -DEFAULT ENTRY$
   ```

   If the user can now perform the operation that caused the linkage fault without generating an error message, the user's private entrypoint search list contains an error. If the user repeats the operation and the linkage fault occurs again, perform Step 2.

2. You should enter the following command to display the entrypoint search list:

   ```
   LIST_SEARCH_RULES ENTRY$
   ```

   Check the displayed list. The rule -PRIMOS_DIRECT_ENTRIES should be at the top. If it is not, check that a file named SEARCH_RULES*>ADMIN$.ENTRY$.SR exists, and that it contains only the rule -PRIMOS_DIRECT_ENTRIES. If not, create that file, or correct it, and try the operation again.

   #### Note

   The LIST_SEARCH_RULES command does not display the SEARCH_RULES*>ENTRY$.SR file, but displays a list stored in memory. The rule -PRIMOS_DIRECT_ENTRIES should appear in the displayed list and in the SEARCH_RULES*>ADMIN$.ENTRY$.SR file, but not in the SEARCH_RULES*>ENTRY$.SR file.

   If the linkage fault persists, a library name may be missing from the SEARCH_RULES*>ENTRY$.SR file. Check that all the libraries necessary to execute the program causing the linkage fault are listed in the ENTRY$.SR file. Add the pathnames of any missing libraries to the end of the file. Check that the list contains no typographical errors and that all the pathnames are correct. If the pathname is for a remote file, check that the line is up, and the disk is added. (It is recommended that target libraries be stored on the local system to improve performance.) If you change the ENTRY$.SR file, perform step 1 again.

SHARED STATIC-MODE LIBRARIES

A system can have a maximum of 32 shared static-mode libraries. Each library is supplied if the customer has purchased that particular software product. See Table 3-6 for a list of shared static-mode libraries. The octal package numbers are used only by the initialization routines that run at cold start.

TABLE 3-6.   Shared Library Package Numbers

Package (octal)	Shared Library
1	Reserved for Prime
2	VKDALB and MPLUSLB
3	Reserved for Prime
4	VFORMS
5	DBMSLB
6	OAS
7	EMACS
10	Reserved for Prime
11	FTS
12	Reserved for Prime
13	PDGS
14	ROAM offline
15	ROAM online
17	Prime INFORMATION
20	PRISAM
21	ESCAPE34
22	OAS
23-24	Prime INFORMATION CONNECTION
25	PRIMEWAY
26	Prime ORACLE
27-37	Reserved for Prime

Features of Shared Static-mode Libraries

Each user of shared static-mode library routines uses space in private segments 6001, 6006, 6007, 6010, and 6011 in addition to the segments otherwise required by programs. These segments are used for the per-user portion of the shared static-mode libraries and represent a reduction in the size of the user's load file but not in the size of the single user working set at run time. These additional segments may be compensated for by a corresponding reduction in the number of segments in the runfile.

Initialization of Shared Libraries

Shared static-mode libraries must be initialized (with the SHARE command) each time the system is cold started. The runfiles containing the libraries to be shared are in the directory SYSTEM.

For each piece of software having static-mode libraries to be initialized, there should be a COMINPUT file in SYSTEM that contains the appropriate SHARE commands. The file PRIMOS.COMI in CMDNC0 should contain the COMINPUT commands to execute the COMINPUT files.

The PRIMOS.COMI file shares memory image files into the proper segments (see Table 3-1) and runs the programs required to inform PRIMOS that shared libraries are activated. After the libraries are initialized, users of programs linked with the special shared library object files may run V-mode and I-mode programs that access these shared libraries. If the shared libraries are not initialized, programs that expect the shared libraries to be resident receive an error message from PRIMOS whenever they attempt to access a shared library routine.

There must be no active users of a static-mode library when that library is being reshared. To ensure this when initializing a shared library, shut down PRIMOS and then reboot it.

Rebuilding and Reinstallation

Each shared static-mode library has a set of runfiles and a command file to install the program. If only one library must be replaced, it is necessary to rebuild that library only. The library command files put all the necessary files into the directory SYSTEM so that installation is easily accomplished by running the appropriate command file.

Caution

Do not reshare a shared static-mode library while it is in use. As programs using the shared libraries execute, links are made to the appropriate shared library routines in such a way that altering the memory image in use by the program can cause random and unpredictable behavior. Changing a shared library has the effect of making such an alteration to the user's memory image. Share new static-mode libraries only when cold starting the system.

It is safe to install the memory image files into SYSTEM at any time because these files are loaded into memory only when the explicit SHARE commands are given (such as during system startup).

ADDING AND MODIFYING SYSTEM SOFTWARE

This chapter contains information on adding and modifying system software:

- Adding your commands to the command directory (CMDNC0)

- Using customer-defined file suffixes

- Changing defaults for compilers

- Adding files to the HELP database

ADDING COMMANDS TO CMDNC0

You can add new commands to the command directory CMDNC0. The commands must be either runfiles (that is, compiled programs linked by BIND or by LOAD) or CPL programs, but they cannot be segment directories. (You can, though, place into CMDNC0 a CPL program that uses SEG to execute a segment directory program.)

Use the COPY command to add the runfile or CPL program to CMDNC0. (You must have at least Add and Use rights to CMDNC0.)

After you install the command, users can invoke it as they would a normal PRIMOS command. For example, if you add a runfile named COMP.RUN to CMDNC0, users entering COMP at the PRIMOS prompt (OK, or ER!) run COMP.RUN, just as entering LD runs CMDNC0>LD.RUN.

For every command you add to CMDNC0, you should create a corresponding HELP file and add it to the HELP* directory. For details, see the section below, *Adding HELP Files.*

Caution

When installing a new version of a CMDNC0 command, it is recommended that you save a copy of the old version in a convenient directory. You can delete the old version after the new version is thoroughly tested and you determine that the old version is no longer needed.

Command Suffixes

Use the following suffixes for your programs in CMDNC0:

● EPFs (V-mode and I-mode runtime programs created with BIND) must end with the suffix .RUN. This suffix is added automatically when an EPF is created with BIND.

● R-mode runtime programs should end with .SAVE. The R-mode loader, LOAD, automatically adds the .SAVE suffix. If you have R-mode programs whose names do not have the .SAVE suffix, you should add the suffix.

● CPL program names must end with the .CPL suffix.

Users do not have to type these suffixes when invoking the commands. If a user types a command, the command processor checks CMDNC0 for files in the following order:

> *command-name*.RUN
> *command-name*.SAVE
> *command-name*.CPL
> *command-name*

Adding CPL Programs

Use COPY to put the CPL program into CMDNC0, as shown below.

```
COPY NEW_PROG.CPL CMDNCO>==
```

A "File in use" or "File open on delete" error message indicates that the current copy of the CPL program in CMDNC0 is being used. Close the file from the supervisor terminal and try again.

After you add the CPL program, any user can invoke NEW_PROG.CPL by typing NEW_PROG at the PRIMOS command prompt.

Adding EPF Programs

With EPFs you need not be concerned with whether a previously existing version of the EPF is in use before putting it in CMDNC0. Compile the program and use the BIND command to link it. (For details on BIND, see the *Programmer's Guide to BIND and EPFs.*)

Copy the EPF runfile into the directory CMDNC0 with the COPY command. COPY notes the existence of the file in CMDNC0, and asks whether you want it replaced. If the file

System Software

in CMDNC0 is in use, COPY changes the name of the old version to *program*.RP*n* (where *program* is the name of the old version and *n* is a number from 0 to 9).

Any user already running the old version of the EPF is never aware of the change, and continues to execute the old version. Any user who now invokes the program gets the new version. You can delete the old version when it is no longer in use.

The following example shows how to replace an in-use EPF:

```
OK, COPY NEW.RUN CMDNCO> = =
EPF file "CMDNCO>NEW.RUN" already exists, do you wish to replace it? YES
New version of EPF file CMDNCO>NEW.RUN now in place.
Old version of active EPF file now named CMDNCO>NEW.RPO.
OK,
```

If there is already a file named NEW.RP0, the old version is renamed NEW.RP1. A subsequent version would create NEW.RP2, and so on, up through NEW.RP9. If all 10 old versions exist and you try to copy an eleventh version into CMDNC0, COPY queries you, as shown in the following example:

```
OK, COPY NEW.RUN CMDNCO> = =
EPF file "CMDNCO>NEW.RUN" already exists, do you wish to replace it? YES
ok to delete EPF file CMDNCO>NEW.RPO? YES
New version of EPF file CMDNCO>NEW.RUN now in place.
Old version of active EPF file now named CMDNCO>NEW.RPO.
OK,
```

If all 10 old versions are in use, the replace operation is not completed, as shown in the following example:

```
OK, COPY NEW.RUN CMDNCO> = =
EPF file "CMDNCO>NEW.RUN" already exists, do you wish to replace it? YES
EPF replace files are all in use.
Unable to replace file. "CMDNCO>NEW.RUN" (qry$del)
OK,
```

If you or your users frequently modify versions of runfiles in CMDNC0, you should delete unused versions from time to time to save space. Use the DELETE command as follows:

```
DELETE CMDNCO>@@.RP(0 1 2 3 4 5 6 7 8 9) -NO_VERIFY -REPORT -FORCE
```

Adding R-mode Programs

R-mode programs can be written only with the FTN compiler and the PMA assembler. To install an R-mode program into CMDNC0, use the COPY command to copy the loaded runfile.

For example, if you have written a utility program called FARLEY.FTN and have compiled and loaded it, copy the program into CMDNC0 as follows:

```
OK, COPY FARLEY.SAVE CMDNCO>==
```

A "File in use" or "File open on delete" error message indicates that the current copy of the program in CMDNC0 is being used. Close the file from the supervisor terminal and try again.

After you add the program, any user can invoke the program by typing FARLEY at the PRIMOS command line.

Disabling Command Line Processing

The You can create commands for which PRIMOS does not process one or more of the usual command line features, which are wildcard options (such as -AFTER), treewalking options (such as -WALK_FROM), and special characters on the command line (such as %, @ and =). For details on these features, see the *PRIMOS Commands Reference Guide.*

Because such commands appear to the user to be the same as standard PRIMOS commands, you must inform users which of your commands perform a nonstandard processing of the command line. The next three sections describe how to change the standard command line processing for a command.

EPF Commands: BIND has built-in subcommands that allow the user to create EPFs that tell PRIMOS whether to process wildcarding, treewalking, iteration, and name generation on the command line. See the *Programmer's Guide to BIND and EPFs* for details.

CPL Commands: PRIMOS processes only iteration for CPL commands. (Iteration is the expansion of lists contained in parentheses.) Wildcards and name generation must be processed explicitly by the CPL program itself. CPL commands are thus processed like the NX$ R-mode commands described in the next section.

R-mode Commands: PRIMOS processes R-mode commands in CMDNC0 in one of three ways:

● If a command name does not begin with either NX$ or NW$, full command processing is done.

● If a command name begins with NW$, iteration and treewalking patterns are processed, but wildcards and name generation patterns are not.

● If a command name begins with NX$, only iteration is processed.

You may want to create or modify commands for which you do not want to use one or more of the command line features. To prevent PRIMOS from performing the standard command line processing for a command, use the following procedure:

1. Rename the command so that it begins with NX$ or NW$.

2. Write a CPL interlude program that accepts the original invocation name and runs the renamed command.

System Software

For example, suppose you want PRIMOS to process only iteration for the command EXEC.SAVE. First, rename the command NX$EXEC.SAVE. Then add to CMDNCO a CPL program named EXEC.CPL, which consists of the following lines:

```
&ARGS ARGS: REST
NX$EXEC.SAVE %ARGS%
&RETURN
```

Users of the old EXEC command can continue to invoke EXEC, which now invokes EXEC.CPL instead. EXEC.CPL passes the unexpanded arguments on to NX$EXEC.SAVE.

Some commands for which you might want nonstandard command line processing are the following.

● Commands added before Rev. 19 that use option names or special characters that conflict with command line features.

● Commands added before Rev. 19 for which you do not want to use one or more of the command line features.

● Commands that perform their own command line processing.

● Commands that you are adding now for which you do not want to use a command line feature. The CPL interlude is not mandatory.

USER FILE SUFFIXES

All file suffixes beginning with the letter U are reserved for customer use. Use this suffix to create classes of user-defined files that are processed by user-written programs and commands.

All filenames must conform to Prime standards. See the *Prime User's Guide* for details on filenames. The following filenames have user file suffixes:

SALES.UDATA
UPDATE.UTRANS
PROBLEMS.UXB

ADDING LIBRARIES

To learn how to create new EPF libraries and add them to your system, read the *Programmer's Guide to BIND and EPFs.* Pay particular attention to the distinction between *program class* and *process class* EPF libraries.

REMOVING UNUSED LIBRARIES

If you plan to remove an EPF library from your system, remember also to remove it from the system's entrypoint search list, SEARCH_RULES*>ENTRY$.SR, and to remove it from any customized entrypoint search lists under your control. Also, suggest to users that they remove it from their customized entrypoint search lists, if any.

CHANGING COMPILER DEFAULTS

You can change the default compiler option settings for F77, Pascal, PL1, PL1G, and VRPG. For a description of how to change the settings and a list of the settings at T1.0-21.0, see the *Translator Family Software Release Document.*

ADDING HELP FILES

System Administrators can add HELP files (on any subject) to the HELP database supplied by Prime. After these site-created HELP files are installed, the PRIMOS HELP command can display them.

The HELP Database

The HELP database contains a collection of files called HELP files. HELP files are text files that contain information about a system facility, a command, or a subsystem. These files are invoked by the HELP command to provide online information about these subjects.

The name of each HELP file consists of two parts: the name of the facility, command, or subsystem, and the suffix .HELP. For example, the HELP file BIND.HELP contains information about the BIND command and subsystem.

System Software

The HELP* Directory

HELP files are kept in the HELP* directory. The HELP* directory also contains the following two text files:

- HELP_INDEX.HELP

- HELP_SEARCH_LIST

The HELP_INDEX.HELP file is a list of all the HELP files in the HELP* directory. If you add your own HELP file to the directory, edit HELP_INDEX.HELP to include the name of the new file. The HELP_INDEX.HELP file is displayed at one of two times: when a user enters the HELP command without an argument; or when the HELP command cannot find an appropriate HELP file and the user answers YES to the command's prompt, Can't find *x*; do you want a list?.

The HELP_SEARCH_LIST file is a list of system-defined abbreviations for commands. This file allows a user to use a standard abbreviation as an argument for the HELP command to view the HELP file for the desired command. For example, typing either HELP CHANGE_PASSWORD or HELP CPW displays the file CHANGE_PASSWORD.HELP.

Creating HELP Files

HELP files are standard ASCII files. To create (or modify) HELP files, use a PRIMOS text editor such as ED or EMACS.

Observe the following two rules when creating HELP files:

- The first three lines of the file are not displayed. You may leave these lines blank, or make them comment lines that indicate the date and author of the file.

- The filename must have the .HELP suffix. (That is, save the file as *command*.HELP.)

Adding Files to the HELP* Directory

Use the following procedure to add HELP files to the HELP* directory:

1. Create the new file with a text editor.

2. File it in HELP*>*command*.HELP (where *command* is the name of the new command).

3. Edit the HELP_INDEX.HELP file to include the new command.

Protecting the HELP Database

When your system is first installed, HELP* is accessible to anyone. You should limit Write access to this directory so that only authorized persons can alter the directory. Set the ACL for the directory to give ALL access (either by name or as a group) to users authorized to alter the database and LUR access to $REST.

PLANNING THE SYSTEM CONFIGURATION

The PRIMOS operating system contains code to manage:

- Access for up to 960 user processes

- Segmented virtual address space for programs up to 64 megabytes per user

- Input/output control

- The file system

- Interactive terminal users and phantom user noninteractive jobs

- Communications systems

PRIMOS is delivered in a single version that configures itself at every cold start. PRIMOS takes its configuration information from a system configuration file that defines system parameters, such as the number of users the system can support and the amount of available physical memory to be used.

Because the details of configuration vary from site to site, you must decide how you want your own system configured. This chapter discusses configuration directives and is intended to help you plan the configuration of your system.

Note

Before creating your system, you should establish a system log book into which you enter the parameters of your system and of your User Profile Data Base. For details on the system log book, see the *System Administrator's Guide, Volume III: System Access and Security.*

THE SYSTEM CONFIGURATION FILE

The system configuration file is usually named CONFIG, and must be in the CMDNC0 directory. It contains a series of configuration directives, one per line.

To establish or change the configuration of your system, you create or modify the system configuration file, and reboot PRIMOS. When the system comes up, the new configuration is in effect.

Read the sections that follow to learn which configuration directives you need. Then see Chapter 6, Configuration Directives, for the details of constructing the configuration file, for directions on starting up PRIMOS without a configuration file, and for a dictionary of the directives.

TYPES OF CONFIGURATION DIRECTIVES

The rest of this chapter presents five general categories of configuration directives:

- Necessary directives, which must be set for the system to function.

- Useful directives, which need not be set, but which, when set correctly, make the system function better.

- Default-changing directives, which do not concern the system but may interest the System Administrator.

- Equipment-specific directives, which are needed if certain equipment is attached to the computer.

- Rarely used directives, which are used for system debugging or which are functionally obsolete. Avoid using these directives. For details on these directives, see Appendix A, Obsolete and Rarely Used Commands and Directives.

All numerical arguments to configuration directives must be octal numbers. Decimal equivalents are provided for ease of calculation.

NECESSARY DIRECTIVES

Six directives must not be omitted from the configuration file:

- COMDEV specifies the command device partition.

- PAGING specifies the paging device partitions.

- SYSNAM specifies the name of the system.

- NTUSR specifies the number of directly-connected terminal users.

- NPUSR specifies the maximum number of phantoms.

- GO marks the end of the configuration file.

Four other directives may be required for networked systems:

- NRUSR specifies the number of remote users via PRIMENET.

- NSLUSR specifies the number of slave processes via PRIMENET.

- NTSUSR specifies the number of Network Terminal Service users.

- NTSASL specifies the number of Network Terminal Service assignable lines.

Command Device

The COMDEV directive specifies which partition is logical device zero, the command device. (This partition is listed first in the output from the STATUS DISKS command.) The CMDNC0 directory on this partition is the one searched when a user invokes an external PRIMOS command.

The argument to COMDEV is the physical device number of the command device partition. See the *Operator's Guide to File System Maintenance* for details on constructing physical device numbers.

Paging Partitions

Each system must have at least one partition reserved for paging. (The paging partition is also referred to as the **paging device** or the **paging disk.**) The system may have up to eight such partitions.

The PAGING directive specifies the paging partitions. This directive must be included in the configuration file.

A paging partition is normally a split disk (that is, it also contains storage space for files). See Chapter 2, Disks and Tape Drives, for important details on paging partitions, split disks, and methods for determining the size of paging partitions.

System Name

The SYSNAM directive specifies the name of the system. The system uses this name to identify itself on any networks to which it may be connected. If your system is already running PRIMENET, you can give it the same name you formerly specified with the -NODE option of START_NET. Otherwise, you should choose a name appropriate for future PRIMENET or NTS operation. The rules for valid system names are on page 6-23.

Even a standalone system requires a system name; PRIMOS will not boot without one. If you omit the SYSNAM directive from the configuration file, PRIMOS will print an error

message, and prompt you for a name. You should enter the name interactively; then remember to add the SYSNAM directive to the configuration file after PRIMOS comes up.

Likewise, if the name in the SYSNAM directive is too long, or contains invalid characters, PRIMOS will print an error message, and prompt you for a correct name.

If the SYSNAM directive is present, but supplies no name, PRIMOS will prompt you for a name, but without printing an error message. You might need this feature in either of two cases:

1. You expect to boot the system with a different name from time to time.

2. You expect to move the disk pack containing the command partition from one machine to another.

Phantom Users: The NPUSR directive sets the number of phantom users. Phantom users can be thought of as users at imaginary terminals because they take their commands from a file rather than from a terminal. You must set the value of NPUSR to at least 4, which is its default value. If you set NPUSR to 3 or less, PRIMOS displays the following message:

```
Warning:  Specified value of NPUSR too small.
Default value (4) will be used.
```

The processes for several services count as phantoms, for the purpose of configuration. When you calculate the value for NPUSR, you should allow phantoms for these services, as appropriate:

● In addition to some servers, which you do not need to configure, DSM uses phantoms (DSMASR) to run applications such as RESUS and SIM. You should allow at least one additional phantom for each application to be used, although the number of extra phantoms DSM actually requires will depend upon which DSM applications you use concurrently.

● Printers require one phantom per active despooler environment.

● Batch service, if used, requires one phantom for the Batch monitor, plus one or more for jobs, up to one for each queue.

● LAN300 network management requires one phantom, and the LHC downline-load and upline-dump share one more.

Note

It is not necessary to configure phantoms for these processes, because they are servers:

Login server	DSM logger
Logout server	Network manager
Timer process	ISC network server
DSM system manager	NTS connection manager
DSM server	Security auditor

● Other communication products, including DPTX, RJE, FTS, and PRIME/SNA™, may also require phantoms.

You may want to configure some phantoms to be available for terminal users. Start with about one phantom for each five terminal users. If your terminal users complain that phantoms are not available, you can increase the number configured. If there are no complaints, you may want to decrease the number configured until there are complaints and then increase it slightly.

If you have PRIMIX on your system, see *Using PRIMIX on the Prime 50 Series* for the number of phantoms you should allot.

Number of Users

The following categories of users and corresponding directives determine how many users your system can support:

● Phantom users, NPUSR (covered above)

● Terminal users, NTUSR

● Network Terminal Service users, NTSUSR

● Remote users, NRUSR

● Slave users, NSLUSR

You can issue a LIST_PROCESS command to display information about each user currently on your system. Remote and slave users are for PRIMENET only. The total number of configured users of all types, NPUSR+NSLUSR+NRUSR+NTUSR+NTSUSR, must be less than or equal to 960 (1700_8) on a 4050, 4150, 6150, 6350, or a 6550. On all other systems this limit is 600 (1130_8).

Terminal Users: The NTUSR directive sets the number of directly-connected terminal users. You can configure up to 512 (1000_8) terminal users. (This number includes the supervisor terminal).

The NTUSR directive, which has no default value, must be included in the configuration file. You must set the directive's value to at least the number of terminals connected to the computer, plus one for the supervisor terminal. Setting the value higher than the number of connected terminals may make it easier to add terminals in the future.

Network Terminal Service Users: The NTSUSR directive sets the number of simultaneous NTS terminal users. Up to 512 (1000_8) are allowed.

Remote Users: Remote users are terminal users on other systems who can log in to your system through their computer, which is networked to yours.   The NRUSR directive, which has a default value of 0, sets the number of remote users.   If you set the value to 0 or omit the directive from the file, no one can log in remotely to your system, regardless of any network connections.   You can allow up to 255 (377_8) remote users on your system.

Slave Users: The NSLUSR directive sets the number of slave users.   Slave users are processes on your system that handle requests (made by users on other systems) for file access, attaching, and so forth.

The default value of NSLUSR is 0.   If you set the value of NSLUSR to 0 or omit the directive from the configuration file, no one can access files on your system from other systems networked to yours.

You can configure up to 255 (377_8) slave users.

You may wish to consult with your Prime System Analyst when setting initial values for remote and slave users.   The required number depends upon your specific network and computers and upon the type of work your users are doing.   Because only 255 virtual circuits are available, it usually does not make sense for NRUSR+NSLUSR to exceed 255.

End of File

The GO directive marks the end of the configuration file.   This directive must be the last noncomment line of the configuration file.   Any subsequent directives will not be acted upon.

USEFUL DIRECTIVES

Useful directives set parameters for utilization of memory, assignable asynchronous lines, and buffers.   (Event logging, no longer controlled by directives, is described in the *DSM User's Guide.*)

Assignable Asynchronous Lines

The NAMLC directive sets the maximum number of directly connected assignable asynchronous lines that can be used simultaneously.   Similarly, the NTSASL directive sets the maximum number of NTS assignable asynchronous lines.   A value of NAMLC or NTSASL that is too low may not become apparent immediately upon cold start.   Assignable asynchronous lines are used by user programs or the spooler to communicate with serial devices such as serial printers.   The default value for both NAMLC and NTSASL is 0. The number of directly connected assignable asynchronous lines (NAMLC) plus the number of directly connected terminal users (NTUSR) cannot exceed 512 (1000_8).

To define a line as assignable, use the SET_ASYNC command with the -ASGN YES option. To change the buffer sizes of an assigned asynchronous line use the CAB command. These two commands are described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

Size of Wired Memory at Cold Start

The WIRMEM directive displays, at the supervisor terminal, the amount of wired memory (in kilobytes) at cold start. Although this value changes during system operation, it provides an indication of the wired memory required to support a particular system configuration.

Number of Locate Buffers

PRIMOS incorporates a memory-to-disk cache that stores the most recently and most frequently accessed disk records, thus reducing disk I/O. This cache is made up of a number of main memory buffers called **locate buffers** (also called **associative buffers**). Each locate buffer is two kilobytes in size. The default number of locate buffers is 64 (100_8).

By using the NLBUF directive, you can configure from 8 (10_8) to 1024 (2000_8) locate buffers. Configuring more locate buffers can decrease disk I/O. However, additional locate buffers can use up more memory, and if not enough memory is available, paging I/O may increase to the point where it cancels the advantage gained by increasing the number of locate buffers. For example, 256 buffers require one-half megabyte of memory.

The optimal number of locate buffers depends upon the applications running on the system. These buffers are most useful when applications access the same file records repeatedly. It may be appropriate to configure more buffers if the USAGE command reports a high locate miss rate (the %Miss field) and a low or normal page fault rate (PF/S). An increase in NLBUF should result in a decrease in %Miss without significantly changing the page fault rate. Also, lowering NLBUF can help where the page fault rate is high, but %Miss is normal or low.

VMFA Dynamic Segments

The NVMFS directive sets the number of VMFA (Virtual Memory File Access) dynamic segments available in virtual address space for the system. VMFA segments are used by EPFs to map segments dynamically.

The default number of VMFA segments is 100 (144_8). You may want to increase this number if users frequently receive such messages as "Not enough segments" or "No space available from process class storage heap," or if you expect them to use a large variety of EPF programs or libraries. You can specify a maximum of 1024 (2000_8) VMFA segments.

If you have PRIMIX on your system, see *Using PRIMIX on the Prime 50 Series* for the number of VMFA segments you should specify.

DEFAULT-CHANGING DIRECTIVES

Default-changing directives change the default values of the directives that control the following: printing the directives as they are being processed, user-defined abbreviations, erase and kill characters, ECCU handling, and certain login and logout procedures.

Displaying Configuration Directives

By default, configuration directives are not displayed at the supervisor terminal as they are processed. To display these directives at the supervisor terminal, include the TYPOUT YES directive in the file.

All directives after the TYPOUT YES directive are displayed until either the TYPOUT NO directive or the GO directive is encountered in the configuration file.

User-defined Abbreviations

By default, users can use the ABBREV command to create abbreviations for PRIMOS commands and their arguments. The abbreviations are stored in abbreviation files. When used on the command line, the abbreviation is expanded by the system's abbreviation processor.

If you do not want users to create and use abbreviations, disable the abbreviation processor by including the ABBREV NO directive in the configuration file.

If you omit the ABBREV directive from the configuration file (or specify ABBREV YES), the abbreviation processor is enabled and users can employ command line abbreviations.

Erase and Kill Characters

Erase and kill characters are used on the PRIMOS command line and within programs. The erase character erases the character to the immediate left of the cursor. For example, typing the word DATE and then next typing the erase character is the same as if you had typed only DAT.

The kill character nullifies all characters to the left of the cursor. For example, typing the word DATE at the OK, prompt and then next typing the kill character is the same as if you had typed nothing after the prompt.

The default systemwide erase character is the double-quote character (") and the default kill character is the question mark (?). To change these default characters, use the ERASE

and KILL directives. If you change either or both of these characters, inform all of your users because Prime's documentation assumes that ERASE and KILL have the original default values. Remember that the character values for erase and kill must be different from one another.

Note

Whether or not the System Administrator changes the default erase and kill characters, users can change these characters for their terminal sessions by using the -ERASE and -KILL options of the PRIMOS TERM command. Details of TERM are given in the *PRIMOS User's Guide.*

ECCU Handling

An ECCU (Error Correction Code Uncorrectable) is a two-bit memory parity error. The MEMHLT directive determines how PRIMOS handles the occurrence of an ECCU.

If the MEMHLT directive is not in the configuration file or if the default MEMHLT YES is included, the system halts when an ECCU occurs.

If MEMHLT NO is in the configuration file and certain conditions are met, PRIMOS can detect what user process encountered the ECCU. PRIMOS then logs out that user process, prints a message at the supervisor terminal listing the user ID of the process, and continues operating normally for other users. See the MEMHLT directive in Chapter 6, Configuration Directives, for the conditions that must be met for the user process to be logged out.

MEMHLT NO is appropriate only if your system is serviced regularly and if you are not running ROAM-based data management products (DBMS, DISCOVER, and PRISAM). If you use MEMHLT NO and your system still halts with memory parity errors, have your system serviced. Otherwise the system may experience an undetectable or falsely corrected error because it is running with faulty memory.

For a discussion of whether to use a warm start or a cold start after a system halt or hang, see the chapter on Equipment and Environment in the *System Administrator's Guide, Volume III: System Access and Security.*

Caution

Systems running ROAM-based data management products (DBMS, DISCOVER, PRISAM) should have MEMHLT YES in the configuration file and should be cold started after any system halt. A cold start is necessary so that rollback of incomplete transactions can occur. A warm start may cause loss of data.

Changing the Login/Logout Procedure

Six directives modify the default login and logout procedure. These directives control the following:

- The printing of login/logout messages at the supervisor terminal (LOGMSG)

- The printing of unsuccessful login messages at the supervisor terminal (LOGBAD)

- The use of the LOGIN command for logged-in users (LOGLOG)

- The automatic logging out of disconnected users (DISLOG)

- The length of the inactivity timeout (LOUTQM)

- The time allowed for a login procedure (LOTLIM)

Printing Login/Logout Messages: When a user logs in or out, a message to this effect is printed by default at the supervisor terminal. These messages provide the System Administrator with a record of these transactions. (Activity of child processes is not recorded.)

If you decide that such detailed information is not necessary, you can disable these messages by using the LOGMSG NO directive in the configuration file. (Disabling such messages saves paper on hard-copy supervisor terminals.) Omitting the LOGMSG directive (or specifying LOGMSG YES) causes these messages to be displayed at the supervisor terminal.

Printing Unsuccessful Login Messages: The LOGBAD directive controls the printing, at the supervisor terminal, of messages about unsuccessful login attempts. If you omit the directive from the configuration file or specify LOGBAD NO, such messages are not printed.

If LOGBAD is enabled (by your having specified LOGBAD YES in the configuration file), any unsuccessful attempt to log in (due to an invalid user ID, incorrect password, or invalid project ID) causes a message to be printed at the supervisor terminal.

Disabling the LOGIN Command: By default, a user can issue the LOGIN command while logged in. (A logged-in user might wish to log in under a different user ID or under a different project.) The user is first logged out and then logged in again according to the arguments of the LOGIN command. External logout and login programs in CMDNC0 are run if they exist.

To allow use of the LOGIN command to logged-in users, either specify LOGLOG YES in the configuration file or omit the directive.

Specifying LOGLOG NO prevents the use of the LOGIN command for logged-in users and forces them to log out explicitly (with the LOGOUT command) before being able to log in again. Forcing users to log out explicitly prevents a user from unknowingly logging out another user who has left a terminal and not logged out.

Logging Out Disconnected Users: The DISLOG directive, described in Chapter 6, provides one means for selecting, for each line, whether disconnection of the line causes automatic logout. The preferred method is to use the SET_ASYNC command with the -DISLOG option, described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers*. The SET_ASYNC command allows you to change the dislog status of lines dynamically.

Use the directives DTRDRP and DISLOG YES or DISLOG *line-number* if you have lines configured for Auto Speed Detect (ASD) so that each line is returned to ASD when the user disconnects.

Inactivity Timeout: You can set the amount of time a terminal can remain idle before its user is automatically logged out (inactivity timeout). Prime supplies a default time of 1000 (1750_8) minutes, which is 16 hours and 40 minutes. To retain this default value for the inactivity timeout, omit the LOUTQM directive from the configuration file.

To change the value of the inactivity timeout, use the LOUTQM directive. For example, LOUTQM 74 sets the inactivity timeout to 1 hour (1 hour equals 60 minutes, whose octal value is 74_8).

Length of Login Procedure: You can use the LOTLIM directive to set the length of time allowed for a user to log in. The default value of three minutes is the recommended length because it gives a user a reasonable amount of time to type in required information without wasting system resources.

To change the time allowed for login, use the LOTLIM directive. The minimum amount of time you can allow is one minute. There is no maximum. The time should always be less than the time allowed by the LOUTQM directive.

Amount of Memory to Use

Omit the MAXPAG directive from the configuration file. MAXPAG limits the amount of memory available to PRIMOS and is not normally useful. One reason for using MAXPAG is to test system performance with reduced memory. Another is to allow PRIMOS to run on a system that has some defective memory in higher addresses. See Appendix A for full information.

EQUIPMENT-SPECIFIC DIRECTIVES

Several directives change parameters that control such equipment as buffers, the AMLC programmable clock, the supervisor terminal, and several types of lines.

Changing Buffer Sizes

Although many devices operate with the default buffer sizes, it is often desirable (and, in some cases, necessary) to change these sizes. At Rev. 22.0 the CAB command, described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers*, replaces the REMBUF, AMLBUF, NTSBUF, and NTSABF configuration directives.

DMC Tumble Tables: The AMLIBL directive sets the size of each input buffer for the AMLC controller DMC (Direct Memory Control) tumble tables. If the directive is not included in the configuration file, the default buffer size is 48 (60_8) halfwords.

An AMLC line attached to a high-speed input device could send data into the tumble tables faster than it could exit, resulting in the loss of the data. In this case, you can increase the buffer size with the AMLIBL directive or you could let the system calculate a value for you. The maximum size for the buffers depends upon the number of controllers and the amount of space available in the system for buffers. To let the system calculate and set this value, specify either AMLIBL 0 or AMLIBL with no argument.

For more information on the DMC tumble tables, see the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

ICS Controllers: The ICS INPQSZ directive changes the size of the input queues for ICS controllers from the default value of 63 (77_8). You may need to change the queue size on systems that have many terminals sending large amounts of data. For further details on configuring ICS lines, see the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

AMLC Programmable Clock

The AMLC hardware contains a software programmable clock. The clock's default baud rate is 9600 (22600_8). To change the default, use the AMLCLK directive with a value in the range from 29 (35_8) to 19200 (45400_8) baud. You should keep the default baud rate if you are using Auto Speed Detect (ASD) on any of your lines.

To specify that an asynchronous line *n* (a decimal number) is to use the programmable clock speed, use the command SET_ASYNC -LINE *n* -DEFAULT -SPEED CLOCK. (SET_ASYNC is a command, *not* a configuration directive.) If you use the -SPEED CLOCK option, but the AMLCLK directive is not in the configuration file, the line assumes the default speed of 9600 (22600_8) baud. See the *System Administrator's Guide, Volume II: Communication Lines and Controllers*, for details on the SET_ASYNC command.

Telephone Lines

The AMLTIM configuration directive sets the values for the three timers associated with dialup lines. You can use the defaults for the first two arguments, *ticks* and *disctime*. However, you may want to set the third timer, *gracetime*. The third timer is essentially the amount of time the system allows a line to remain active without a process being logged in.

A reasonable value for *gracetime* is from 1 to 3 minutes. The value for one minute (which is 600 tenths of a second) is 1130_8 and is specified with the AMLTIM directive as follows:

```
AMLTIM 2 3410 1130
```

(The first two arguments, 2 and 3410, are the defaults for the first two timers.) To set *gracetime* to two minutes, use the value 2260_8. To set *gracetime* to three minutes, use the value 3410_8.

When the user logs out, the line remains active for the period specified by the *gracetime* argument. If the user was logged in on a dialup line and hangs up the telephone without logging out, whether the DTR (Data Terminal Ready) signal is dropped depends on the presence or absence of the DTRDRP directive in the system configuration file.

- If DTRDRP is not in the configuration file, the DTR signal is dropped within the time period specified by the first argument (*ticks*) to AMLTIM, causing the line to become inactive.

- If DTRDRP is in the configuration file, the DTR signal is dropped immediately, regardless of the value of *gracetime*. This prevents a user from dialing in and being connected (with full access rights) to the process of another user who has disconnected without logging out.

Supervisor Terminal

Two directives, ASRATE and ASRBUF, change the default baud rate and buffer size of the supervisor terminal. You may have to use these directives if you have a nonstandard supervisor terminal. For details on using them, see Chapter 6.

Changing the Baud Rate: The ASRATE directive sets the baud rate of the supervisor terminal. Many hard-copy supervisor terminals have a baud rate of 300, which is the default of ASRATE, although some can use 1200. The rate of 110 baud is available for slower terminals.

If you have a screen terminal (such as a PST 100™ or a PT200™) as your supervisor terminal, you may want to use one of the other available baud rates of 1200 or 9600.

If used, the ASRATE directive should be the first directive in the configuration file.

Changing Buffer Sizes: Most hard-copy supervisor terminals can use the default input buffer of 256 (400_8) bytes and the default output buffer of 384 (600_8) bytes.   If your supervisor terminal runs at greater than 300 baud, or if you are using LAB or DSM SIM commands, you might need to use the ASRBUF directive to increase the output buffer size.

If you are using the RESUS remote supervisor terminal facility, your main and remote supervisor terminals may be operating at different baud rates.   Because the data stream is sent to *both* terminals, the faster terminal may occasionally be held to the speed of the slower unless you increase the buffer size of the *slower* terminal.

Synchronous Lines

Synchronous lines to other computers and devices are enabled and configured with the SYNC directives.   Which directive is used and what values are assigned depend upon the specific hardware and controllers on your system.   The SYNC ON directive is used for all synchronous line types, including MDLC and ICS lines.

For details on the four SYNC directives, see Chapter 6, Configuration Directives.

Note

At Rev. 20, SYNC became a synonym for SMLC.   For example, specifying SYNC ON is the same as specifying SMLC ON.   SYNC is now the preferred term.

Uninterruptible Power Supply

An Uninterruptible Power Supply (UPS) maintains power to the CPU and memory during a power failure.   On many systems it also automatically performs a warm start.   If your system has UPS, the UPS directive determines what action is taken after a warm start.

The UPS directive with an argument of 0 produces a warm start followed by a halt.   The operator must then intervene to bring up the system.

The UPS directive with a positive argument tells UPS to perform a warm start and then wait for a number of seconds (as specified by the argument) before bringing up the system. The delay allows the disks to reach full speed before PRIMOS attempts to access them.   For example, UPS 100 tells UPS to wait 64 seconds after a warm start before it brings up PRIMOS.   A value of 100_8 is recommended for a storage module.   If your system does not have an Uninterruptible Power Supply, omit the UPS directive from the configuration file.

ICS Controllers

If your system has an ICS controller, you may use these two directives:

- ICS INTRPT, which sets the asynchronous interrupt rate

- ICS CARDS, which verifies an asynchronous Line Adapter Card configuration of an ICS2 or ICS3 controller

● ICS INPQSZ, which sets the size of the input queue buffers for ICS lines.

In addition, the ICS INPQSZ directive sets the size of the ICS input queue buffers. This directive is described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

Modifying the Interrupt Rate: To set a faster interrupt rate than the default 100 millisecond rate on lines connected to ICS controllers, use the ICS INTRPT directive. You can set the rate to a value between 100 ms and 10 ms. (To set the interrupt rate on lines connected to AMLC controllers, use the AMLC command, as explained in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*) It is usually better to change the buffer size for line for a particular device than to change the interrupt rate.

Verification of ICS Configuration: ICS2 and ICS3 controllers, which can have up to 16 Line Adapter Cards (LACs), are configured at each cold start. To check that the LAC configuration is as you expected, include the ICS CARDS directive in the configuration file.

If the actual LAC configuration at cold start is different from that specified by this directive, PRIMOS displays an error message explaining the discrepancy. Such a discrepancy can occur if an extra LAC was added since you configured your system, or if a LAC went bad since the system was last cold started.

If the ICS CARDS directive is omitted for an ICS2 or ICS3 controller, that controller's configuration is not checked at cold start.

For further details on the ICS CARDS directive, see Chapter 6, Configuration Directives, or the *ICS User's Guide.*

CONFIGURATION DIRECTIVES

The PRIMOS operating system configures itself at every cold start. The configuration information that PRIMOS needs is stored in the system configuration file. The first half of this chapter discusses this file and the PRIMOS CONFIG command that processes it. The second half presents all configuration directives alphabetically, explaining each directive and its arguments.

SYSTEM STARTUP AND CONFIGURATION

The System Startup File

At cold start, PRIMOS runs the **system startup file**, PRIMOS.COMI, in the CMDNC0 directory. The first command in that file must be the CONFIG command, which specifies the system configuration file and begins its processing.

PRIMOS.COMI also must contain several other commands for initializing communication lines and various software packages. See the *System Administrator's Guide, Volume II: Communication Lines and Controllers* for details.

Notes
The name C_PRMO for the system startup file is obsolete, but still supported. If your system uses C_PRMO, you should change it to PRIMOS.COMI.

If CMDNC0 contains both PRIMOS.COMI and C_PRMO, PRIMOS uses PRIMOS.COMI.

The *Rev. 22.0 Software Installation Guide* contains a few annotated samples of startup and configuration files. A template of the PRIMOS.COMI system startup file is shipped with each Prime system, as PRIRUN>PRIMOS.COMI.TEMPLATE. Copy it to CMDNC0 as PRIMOS.COMI, and use a text editor to change the template to suit your installation.

The CONFIG Command

The CONFIG command

functions only during system startup. The format of the CONFIG command is:

   CONFIG -DATA *config-filename*

config-filename is the name of the configuration file in the CMDNC0 directory. In the following example, the current directory is CMDNC0 and the name of the configuration file is CONFIG:

```
CONFIG -DATA CONFIG
```

Note

If you reexecute the PRIMOS.COMI startup file while PRIMOS is running (by issuing a **COMINPUT PRIMOS.COMI** command at the supervisor terminal), the file CMDNC0>CONFIG.CPL executes, preventing the generation of an error when the system encounters the CONFIG command in the startup file. Instead, CONFIG.CPL displays the warning

```
Primos already running, CONFIG command is ignored.  (CONFIG.CPL)
```

and PRIMOS continues executing the PRIMOS.COMI file. (You may occasionally need to reexecute the startup file to re-share and reinitialize system software.)

THE SYSTEM CONFIGURATION FILE

Because PRIMOS is configured each time the system is cold started, the System Administrator can reconfigure a system as necessary to meet changing needs. Most systems, however, use a constant set of configuration directives. The directives that configure the system are stored in a text file called the **system configuration file**. The system configuration file is usually named CONFIG and must be stored in the CMDNC0 directory.

Creating the Configuration File

Because configuration files vary greatly from system to system, a template configuration file is not shipped with the system software. Therefore, it is your responsibility to create the configuration file.

Use the following rules and guidelines when creating the configuration file:

● Boot the system without a configuration file (see page 6-4), and, once the system has booted, use the NSED text editor, under PRIMOS, to create an initial configuration file.

● Enter each configuration directive on a separate line. The directives may be entered in uppercase or lowercase.

- You may insert comment lines after a directive or on a separate line. Comments must begin with the /* character pair.

- The ASRATE command should normally be the first directive in the file

- The COMDEV, PAGING, NTUSR, NPUSR, SYSNAM, and GO directives must be in the configuration file. GO must be the last directive in the file.

- All numerical values in the configuration file must be in octal. For details on constructing physical device numbers needed as arguments for certain directives, see the *Operator's Guide to File System Maintenance.*

To change the system configuration for the next cold start, modify the configuration file with a text editor. The next time the system is brought up, the new configuration takes effect. If you cannot bring up PRIMOS because of errors in the configuration file, use the procedure described on page 6-4 in the section Booting PRIMOS Without a Configuration File.

Network Information

If your system will be part of a PRIMENET or NTS network, you must set certain configuration directives, such as NSLUSR, NRUSR, or NTSUSR. Other information dealing with the computer's interface to the network is stored in a global network configuration file.

For PRIMENET, the System Administrator creates the global network configuration file with the CONFIG_NET utility. The file is usually stored in the PRIMENET* directory.

For additional PRIMENET information, see the *User's Guide to Prime Network Services* and the *Operator's Guide to Prime Networks.* For information on configuring and administering the PRIMENET system, see the *PRIMENET Planning and Configuration Guide.*

For NTS, the System Administrator creates the global network configuration file with the CONFIG_NTS utility. The file is usually stored in the NTS* directory.

For additional NTS information, see the *NTS User's Guide.* For information on configuring and administering NTS, see the *NTS Planning and Configuration Guide.*

Examining the System Configuration

To examine the configuration of a running PRIMOS system, use the LIST_CONFIG command. For more information about LIST_CONFIG, see the *DSM User's Guide.*

Configuration File Errors

Certain errors in the configuration file prevent a successful cold start of the system. When an unsuccessful cold start occurs, a message is displayed at the supervisor terminal that explains which configuration directive was at fault and requests a system restart. You can then boot PRIMOS without a configuration file (as described below), correct the file, and reboot PRIMOS.

The most common causes of errors are:

- Placement of configuration directives into the system startup file (PRIMOS.COMI or C_PRMO).

- Placement of system startup commands into the configuration file.

- Decimal numbers used by mistake instead of octal numbers.

- Out-of-range values for parameters (for example, a value of 12 in the SYNC directive, which only takes values from 00 to 07).

- Values that are correct in themselves, but conflict with other values. For example, the value for NPUSR+NSLUSR+NRUSR+NTUSR+NTSUSR, on the 4050, 4150, 6150, 6350, or 6550 processor, must be no greater than 1700_8 (decimal 960). Values of 760_8 for NPUSR and NTUSR, plus 20_8 for NRUSR and NSLUSR, each correct in isolation, produce a sum ($760_8+760_8+20_8+20_8=2000_8$) greater than 1700_8 and prevent the system from starting up.

For a listing of PRIMOS cold-start messages, including those produced by erroneous directives, see your CPU handbook.

Booting PRIMOS Without a Configuration File

To boot PRIMOS without a configuration file, use the manual boot procedure. See the CPU handbook for your computer model for detailed information on booting. Brief instructions are given below.

These three situations may required manual booting:

- The system is new and has no configuration file.

- The configuration file or PRIMOS.COMI has been damaged or made incorrect.

- The hardware configuration has been changed, and the configuration file now is inappropriate for booting. (For example, COMDEV is now on a different disk drive or controller.)

Caution

Avoid using the obsolete single-user system PRIMOS II. In particular, do not use NSED under PRIMOS II to edit the configuration file on a Rev. 20 or later disk. PRIMOS II cannot write to Rev. 20 or later disks. Instead, use NSED under manually-booted PRIMOS, as explained below.

To change the configuration file using the manual boot, perform the following five steps:

1. At the CP> prompt, use the SYSCLR command to master-clear the system.

2. At the next CP> prompt, enter the BOOT command with the 100000 boot option added to the option word (for example, BOOT 14134 becomes BOOT 114134). Respond to the prompts:

`Enter COMmand DEVice:`	The physical device number for the command partition. This is the argument to the COMDEV directive.
`Enter PAGing DEVice:`	The physical device number for a paging partition. This must be a single value, as for the now-obsolete PAGDEV directive, and *not* the multiple values permitted by the PAGING directive.
`Enter Number Terminal USeRs:`	Any appropriate small number. Because the system is being brought up only to change the configuration file, there will be no users other than the supervisor terminal. This number is the argument to the NTUSR directive.

Coldstart now begins, and a few messages appear, followed by another prompt.

3. Respond to the prompt:

`Enter system name:`	The six-character name for this system. This is the argument to the SYSNAM directive.

A few more messages now appear as PRIMOS comes up. The PRIMOS.COMI file is *not* executed, with the result that no user software is shared, and no users other than the supervisor terminal are permitted.

4. Use NSED (the nonshared editor) to edit the configuration file.

5. Try out the new or revised configuration by shutting down PRIMOS, and rebooting it without the 100000 boot option.

Notes

Adding the 100000 value to the BOOT option word (at Step 2) instructs PRIMOS to ignore the PRIMOS.COMI system startup file. As a consequence, PRIMOS does not read the configuration file. Instead, PRIMOS queries the operator for the COMDEV, PAGDEV, NTUSR, and SYSNAM parameters, and uses default values for the other configuration parameters.

PRIMOS also does not downline-load the communication controllers or any intelligent disk controllers.

Because DSM has not started, no system logging takes place.

CONFIGURATION DIRECTIVES

This section describes the directives used in configuration files. See also Chapter 5, Planning the System Configuration, for more information and guidelines on these directives.

▶ ABBREV $\begin{bmatrix} YES \\ NO \end{bmatrix}$

Controls user abbreviations.

YES enables the abbreviation processor, allowing use of the ABBREV command. YES is the default.

NO prohibits the use of the ABBREV command.

▶ AMLBUF *line* [*in-buff-size* [*out-buff-size* [*dmq-size*]]]

Sets the size of an asynchronous line's I/O buffers.

The AMLBUF directive, obsolete at Rev. 22.0, is described in Appendix A. To set buffer sizes, use the CAB command, described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers*.

▶ AMLCLK *baudrate*

Sets the software programmable clock in the AMLC hardware to a baud rate of *baudrate* bits per second.

Parameter	Minimum	Maximum	Default	Recommended
baudrate, octal (decimal)	35 (29)	45400 (19200)	22600 (9600)	as required

The value specified for *baudrate* must be no less than 35_8 (29 decimal) and no greater than 45400_8 (19200 decimal). The default is 22600_8 (9600 decimal).

The default speed is recommended if you are using Auto Speed Detect (ASD) on any line.

When used on a system where ICS or LHC300 asynchronous lines are present, *baudrate* must be one of the valid baud rates listed in the table under the ASYNC JUMPER directive.

▶ AMLIBL *buffer-size*

Sets the size of the DMC AMLC input tumble tables at cold start.

buffer-size is the number of halfwords allocated to each input buffer. Except for the special value of 0 described below, *buffer-size* must be greater than 20_8. The maximum value of *buffer-size* is variable and depends on the number of AMLC controllers configured and the amount of space available in the system for buffers.

If *buffer-size* is 0 or is omitted, the size of the buffers is automatically calculated as the maximum size allowed by the available buffer space. If the AMLIBL directive is omitted from the configuration file, the default buffer size is 60_8 (decimal 48).

Each AMLC controller has one pair of buffers and all buffers are configured to the same size. Data is stored one character per halfword.

During cold start initialization, the error message BAD AMLIBL PARAMETER (CINIT) is displayed if *buffer-size* is too small, and INPUT BUFFERS TOO LARGE (AMINIT) is displayed if *buffer-size* is too large. Modify the parameter to be a value within the permissible range as described above.

See the *System Administrator's Guide, Volume II: Communication Lines and Controllers*, for more information on configuring asynchronous lines.

▶ **AMLTIM** [*ticks* [*disctime* [*gracetime*]]]

Sets time intervals for the three variable event timers.

Parameter	Minimum	Maximum	Default	Recommended
ticks, octal	1	-	2	24 or more
disctime, octal	0	-	3410	400 or more
gracetime, octal	0	-	0	1130

The arguments have the following values and meanings:

ticks
The interval (in tenths of a second) between carrier check operations. At the end of each period, PRIMOS checks each line for carrier loss. If a loss has occurred and the DISLOG directive is set, the process is logged out. The value for *ticks* must be greater than 0. The default is 2 (0.2 seconds).

disctime
The time period (in tenths of a second) when the DTR signal is again raised on all lines. A modem that was forced to be inactive because the DTR signal was dropped will again be able to establish an active carrier. (The most common cause for a carrier to be inactive is that a dialup line has been hung up.) Specifying a value of 0 disables this feature. Otherwise, the value must not be less than the value of *ticks* and is truncated to the nearest multiple of that value. The default is 3410_8 (1800 decimal, which is 3 minutes).

gracetime   The minimum grace period (in tenths of a second) for terminal lines that have active carriers but are not connected to processes that are logged in. *gracetime* in effect defines the minimum time for a caller to establish itself with a logged-in process. (The actual grace period varies from *gracetime* to twice *gracetime*.) The default value of 0 disables the grace period. The specified value (if not 0) must be greater than *ticks* and is truncated to the nearest multiple of *ticks*. The value of *gracetime* should be large enough to enable PRIMOS to generate a forced logout of a previous user and enable another user to complete a login attempt.

Notes

The AMLTIM directive affects the operation of Auto Speed Detect (ASD). No standard settings for the AMLTIM parameters can be recommended if your installation has ASD, but the following values for *ticks*, *disctime*, and *gracetime* have been shown to give satisfactory results. Set the value for *ticks* to at least 24_8. Set the value for *disctime* to at least twice that of *ticks*, preferably larger than 400_8. Set a value of 1000_8 for *gracetime*. ASD uses up a portion of *gracetime* before a user logs in.

For NTS terminal lines there is no carrier signal. The value specified for *disctime* is ignored, and *gracetime* is the time allowed for an LTS user to log in, once connected.

▶ **ASRATE** *rate*

Sets the baud rate of the supervisor terminal.

The only valid values for *rate* are the following four octal integers:

Octal code	Baud rate (decimal)
0110	110
1010	300 (default)
2010	1200
3410	9600

Note

The values for *rate* are coded bit patterns, and are **not** the octal equivalents of decimal values.

If used, ASRATE should be the first directive in the configuration file because it ensures that any subsequent configuration error messages are displayed at the desired speed.

▶ **ASRBUF** 0 [*in-buff-size* [*out-buff-size*]]

Sets the sizes of the supervisor terminal I/O buffers.

Parameter	Minimum	Maximum	Default	Recommended
in-buff-size, octal	0	7777	200	200
out-buff-size, octal	0 or 100	7777	300	300, but see below.

The arguments have the following values and meanings:

in-buff-size   The size in halfwords (two characters per halfword) of the supervisor terminal input buffer. The default is 200₈ (128 decimal). If 0 is specified, the buffer size remains at its previously set value (which is usually the default size).

out-buff-size   The size in halfwords (two characters per halfword) of the supervisor terminal output buffer. The default is 300₈ (192 decimal). The minimum value (other than 0) is 100₈ (64 decimal). If 0 is specified, the buffer size remains at its previously set value (which is usually the default size).

If you are using screen-formatted commands such as LAB, DSM SIM, or USAGE at the supervisor terminal, you should increase *out-buff-size* to at least 2000₈ (1024 decimal).

You might also want to enlarge *out-buff-size* if you have the following situation: a 300 or 1200 baud supervisor terminal, LOGMSG enabled (which prints login and logout messages at the supervisor terminal), and many logins and logouts. Such a situation can result in noticeable delays if frequent messages are sent to a supervisor terminal with a standard *out-buff-size* and a relatively slow baud rate.

▶ **ASYNC JUMPER** *speeda speedb speedc*

Defines the available line speeds for asynchronous lines.

speeda, *speedb*, and *speedc* are line speeds (specified as bits per second) in octal. These three speeds can be chosen from the list below. The speeds you can use on lines configured for Auto Speed Detect (ASD) are marked with an asterisk. ICS lines and NTS assignable lines support all of the speeds listed below.

Speed (bps)	*Octal Value*	*Speed (bps)*	*Octal Value*
50	62	1800	3410
75	113	2400	*4540
110	*156	3600	7020
150	226	4800	*11300
200	310	7200	16040
300	*454	9600	*22600
600	*1130	19200	*45400
1200	*2260		

The defaults are 75 for *speeda*, 150 for *speedb*, and 1800 for *speedc*.

Use the following guidelines in determining whether to specify the ASYNC JUMPER directive:

● If you have only unmodified AMLC lines and are not using ASD, do not use the ASYNC JUMPER directive because it does not affect the lines.

- If you have an AMLC board on which hardware jumpers provide non-default line speeds on that AMLC, you must use ASYNC JUMPER for those speeds, regardless of whether you are using ASD.

- If you have AMLC lines and are using ASD, use the ASYNC JUMPER directive and specify speeds that match the speeds of the on-board hardware jumpers. Line speed is determined by both the on-board hardware jumpers and by this directive.

- If you have both AMLC lines and ICS lines, use the ASYNC JUMPER directive to set both types of lines to the same value. What speed you choose is up to you, but the speed for both types of lines must be the same, and must match the hardware jumper speeds on the AMLC boards.

▶ **COMDEV** *pdev*

Specifies *pdev* as the physical device number of the command device, the partition on which the system command directory CMDNC0 resides. See the *Operator's Guide to File System Maintenance* for details on the construction of *pdev* numbers.

The partition specified by this directive becomes logical disk zero. The COMDEV directive must be specified in the configuration file.

▶ **COMDVM** *pdev*

Specifies *pdev* as the physical device number of the partition that mirrors the command device (COMDEV) and enables mirroring of file system partitions.

This directive turns on command device mirroring at system startup. If the COMDVM directive is used, it must follow the COMDEV directive in the configuration file.

▶ **DISLOG** $\begin{bmatrix} \text{YES} \\ \text{NO} \\ \textit{line-number} \end{bmatrix}$

Enables automatic logout when a line is disconnected. A line is defined as disconnected when the carrier detect signal goes logically low.

Note

To set DISLOG dynamically on individual lines, you can use the -DISLOG and -NO_DISLOG options of the SET_ASYNC command, which is described in the *System Administrator's Guide Volume II: Communication Lines and Controllers*. This method is preferable to using the DISLOG directive.

DISLOG YES enables automatic logout-on-disconnect for all users. DISLOG *line-number* enables it only for the specified line. (You can specify only one line number per each DISLOG directive, but you can specify as many DISLOG directives as you need. The value of *line number* may be up to 1000_8, which is 512 decimal.) DISLOG NO serves as an

indication that automatic logout-on-disconnect is not desired, but neither enables nor disables it for any line. NO is the default.

Automatic logout is useful for installations with port selectors or dialup modems. Use the SET_ASYNC -DISLOG command, or specify DISLOG YES or DISLOG *line-number*. In addition, specify DTRDRP if you are using Auto Speed Detect (ASD).

DISLOG YES overrides all other DISLOG directives. DISLOG YES and DISLOG *line-number* override DISLOG NO. DISLOG NO cannot override anything. The relative position of multiple DISLOG directives within the configuration file is unimportant. To disable a DISLOG directive, you must comment it out, or remove it from the file.

NTS lines automatically have logout-on-disconnect, regardless of DISLOG.

▶ **DTRDRP**

Controls the dropping of the DTR (Data Terminal Ready) signal associated with an asynchronous line.

If specified, the DTRDRP directive automatically forces the dropping of the DTR for any user when that user logs out, regardless of the period set by the *gracetime* value of the AMLTIM directive.

DTRDRP is useful only for installations using Auto Speed Detect (ASD) with port selectors or dialup modems. (Users who have logged out can also issue the PRIMOS DROPDTR command explicitly.)

▶ **ERASE** $\begin{bmatrix} character \\ octal\text{-}value \end{bmatrix}$

Sets the system default erase character. *character* is any printing ASCII character. *octal-value* is the octal value of any ASCII character.

The ERASE directive accepts either a printing character or an octal number as its argument. In this example, either of the following directives sets the system default erase character to the exclamation mark (!):

    ERASE !
    ERASE 241

If you omit the ERASE directive, the default erase character is the double quotation mark ("), which is 242_8. The only two characters normally recommended as the erase character are (") and backspace. Use ERASE 210 to set the erase character to the BACKSPACE key.

The ERASE and KILL directives take effect upon being processed (rather than later in the startup), with the following results:

- If the SYSNAM directive is omitted from the configuration file, the operator may use the specified erase and kill characters to correct errors when answering the "Enter system name" prompt.

- If the SYSNAM directive is present, but PRIMOS prompts the operator because SYSNAM does not have a valid argument, the specified erase and kill characters are in effect only if the ERASE and KILL directives come before the SYSNAM directive in the configuration file.

- In the unlikely event that an erase or kill character is used in a later directive in the configuration file, it erases or kills on that line in the file. For example, the following directives would result in a system name of SY3, rather than SYS#3.

```
ERASE #
SYSNAM SYS#3
```

▶ GO

Marks the end of the configuration file.

Any subsequent lines are ignored after this directive. The configuration file must include the GO directive as the last noncomment line of the file.

▶ ICS CARDS *device-address config-word*

Checks the asynchronous Line Adapter Card (LAC) configuration of an ICS2 or ICS3 controller.

The ICS CARDS directive causes PRIMOS to check the actual configuration of an ICS2 or ICS3 controller at both cold and warm starts. An error message is displayed if the actual configuration differs from that specified by this directive. (For these error messages, see the *ICS User's Guide* or the handbook for your particular computer model.) The differences are due to unexpected, faulty, or missing LACs.

Whether or not you use the ICS CARDS directive, ICS2 and ICS3 asynchronous configurations are maintained from cold start to shutdown (including across warm starts). However, you should use the ICS CARDS directive to determine that the ICS2 and ICS3 asynchronous configurations have not changed.

If the ICS CARDS directive is omitted for any ICS2 or ICS3 on your system, its configuration is not checked at cold start.

The arguments have the following values and meanings:

device-address   The address of the ICS2 or ICS3 controller. Valid values are 10, 11, 15, 16, 17, 32, 35, 36, 37, 52, 53, and 54.

config-word   The octal conversion of a 16-bit word in which each bit represents a slot in the LAC Card Cage. A bit with a value of 1 means that an asynchronous LAC is present in that slot. A bit with a value of 0 means that either a synchronous LAC is present in that slot or the slot is empty. The *System Administrator's Guide Volume II: Communication Lines and Controllers* explains how to calculate *config-word*.

For further details on ICS CARDS, see the *ICS User's Guide*.

▶ **ICS INPQSZ** *queuesize*

Sets the size of all the ICS input queues.

Parameter	Minimum	Maximum	Default	Recommended
queuesize, octal	17	1777	77	77

The default size of the ICS input queues is 77 halfwords (63 decimal). Data is stored one character per halfword.

queuesize, which is the octal length of the queue, must be no greater than 1777 and must be equal to one less than a power of two. Examples of possible queue sizes are 17, 37, 77, 177, 377, 777, and 1777.

Specifying an invalid value causes cold start to fail and an appropriate error message to be displayed.

See the *System Administrator's Guide, Volume II: Communication Lines and Controllers*, for further details on configuring ICS lines.

▶ **ICS INTRPT** *rate*

Sets the async interrupt rate for ICS controllers.

Parameter	Minimum	Maximum	Default	Recommended
rate, octal	12	144	12	See below.

rate is an integer specifying the number of interrupts per second. The default and minimum values are both 12 (10 decimal), representing a 100-millisecond interrupt rate. The maximum value is 144 (100 decimal), representing a 10-millisecond interrupt rate.

Calculating Interrupt Rates: To set a value between 12 and 144, either use the table below or divide 100 by the desired interrupt rate in milliseconds and truncate the result. Then convert the result (which equals the number of 10-millisecond intervals between interrupts) into octal.   The interrupt rate must be in multiples of 10 milliseconds.   The *Equivalent Asynchronous Baud Rate* column in the table shows what the last asynchronous line would be set at for equivalent performance.

Number of Interrupts per Second		Interrupt Period (milliseconds)	Equivalent Asynchronous Baud Rate
Octal	Decimal		
<12	<10 (error)	100	100
12	10	100	100
13	11	90	111
14	12	80	125
15-16	13-14	70	143
17-20	15-16	60	167
21-24	17-20	50	200
25-31	21-25	40	250
32-41	26-33	30	333
42-62	34-50	20	500
63-144	100	10	1000
>144	>100 (error)	10	1000

Errors: If you specify a value for *rate* that is less than 12, the default rate of 12 is used.   If you specify a value that is greater than 144, *rate* is set to 144.   In either case, the error message BAD ICS DIRECTIVE: INTRPT is displayed and cold start continues.

▶ **KILL** $\begin{bmatrix} character \\ octal\text{-}value \end{bmatrix}$

Sets the system default kill character.   *character* is a printing ASCII character.   *octal-value* is the octal value of any ASCII character.

The KILL directive accepts either a printing character or an octal number as its argument. In this example, either of the following directives sets the system default kill character to the question mark (?):

```
KILL ?
KILL 277
```

If you omit the KILL directive, the default kill character is the question mark (?), which is 277. This is the character usually recommended as the system default kill character. In some circumstances another character, such as DEL (also known as RUBOUT), may be appropriate. Use KILL 377 to set the system default kill character to the DEL key.

The ERASE and KILL directives take effect upon being processed (rather than later in the startup). For more information, see the ERASE directive, on page 6-11.

▶ LHC *number address*

Sets the physical address assignments for LAN Host Controllers to agree with their physical location in the backplane. The most recent LHC directive overrides any existing address assignments.

The arguments have the following values and meanings:

number   Indicates the logical number assigned to the LHC in the NTS configuration file. *number* ranges from 0 to 7

address   Specifies the LHC's physical device address in octal. Valid addresses are 10, 11, 15, 16, 17, 32, 35, 36, 37, 50, 51, 52, 53, 54, and 56, provided these locations are not occupied by another controller.

Use the LIST_COMM_CONTROLLERS command to display the current logical controller number and the octal device address. The two-digit octal device address can be obtained with the STATUS COMM command as well.

See the *System Administrator's Guide Volume II: Communication Lines and Controllers*, for more information on the LHC directive.

▶ LOGBAD $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

Enables the printing of messages about unsuccessful login attempts.

If LOGBAD is enabled with the YES argument, any login attempt that is unsuccessful (due to an unrecognized user ID, incorrect password, or incorrect project) causes a message to be displayed on the supervisor terminal.

The default is NO, which does not cause messages to be displayed at the supervisor terminal about unsuccessful login attempts.

▶ LOGLOG $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

Allows the use of the LOGIN command while logged in.

YES specifies that users can use the LOGIN command while logged in. If a user logs in on a terminal that already has a logged-in user, the logged-in user is first logged out and then the new user is logged in. YES is the default.

NO specifies that the LOGIN command is inhibited for a logged-in user.

▶ **LOGMSG** $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

Enables the display of login and logout messages.

YES specifies that a message be displayed at the supervisor terminal when a user logs in or logs out. YES is the default.

NO specifies that login and logout messages are suppressed.

If you use LOGMSG, have many users logging in and out frequently, and have your supervisor terminal running at 300 or 1200 baud, you may want to enlarge the output buffer size of the supervisor terminal to increase its efficiency. See the ASRBUF directive on page 6-8.

▶ **LOTLIM** *minutes*

Specifies login time limit in minutes.

Parameter	Minimum	Maximum	Default	Recommended
minutes, octal	1	none	3	3

minutes is the octal number of minutes allowed for a user to log in. The minimum value is any value greater than 0. The default is three, which is recommended for most systems, because it gives users adequate time to type, and prevents wastage of system resources. There is no maximum value for *minutes*, but the value should be less than the time allowed by LOUTQM.

▶ **LOUTQM** *minutes*

Specifies inactivity time for automatic logout.

Parameter	Minimum	Maximum	Default	Recommended
minutes, octal	2	none	1750	144

minutes is the number of minutes of inactivity (minus 1) allowed at the terminal before the user is automatically logged out. The value should be greater than 1. The default is 1750_8 (1000 decimal minutes, which is 16 hours and 40 minutes).

▶ MEMHLT $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

Controls the handling of an ECCU (Error Correction Code Uncorrectable), which is a two-bit memory parity error.

YES, which is the default, halts the system when an ECCU occurs.

If NO is specified and certain conditions are met, PRIMOS determines which user process encountered the ECCU, logs out that process, displays a message at the supervisor terminal listing the ID of the process, and continues normal operation. The form of the message is as follows:

```
User 48 (NETMAN) logged out due to a memory parity error.
```

The following conditions must be met for the single failing user process to be logged out.

● The process that is running must not be User 1.

● The system must not be in process exchange (switching between processes).

● The process that is running must be executing in Ring 3.

● The page that took the ECCU must not be wired.

● If the page that took the ECCU is shared by more than one user, an up-to-date copy of the page must be on disk.

Caution

Systems running ROAM-based data management products (DBMS, DISCOVER, PRISAM) should have MEMHLT YES in the configuration file and should be cold started after any system halt. A warm start may cause loss of data.

▶ MTRS *max-size*

Specifies the maximum tape record size allowed on the system.

Parameter	Minimum	Maximum	Default	Recommended
max-size, octal	14000	40000	14000	As required.

max-size specifies the largest magnetic tape record size, in halfwords, that the system can handle. The smallest value, 14000_8, is 6144_{10} halfwords, or about 12 kilobytes. The largest value, 40000_8, is 16384_{10} halfwords, or about 32 kilobytes.

The increased maximum record size allows programs, via the T$MT call, to perform tape I/O with tapes made on other systems that use a greater record size than was supported by pre-Rev. 22.0 PRIMOS.

If the value supplied to the MTRS directive is not within the specified range, the default value of 14000 is used and an error message appears during cold start:

```
Value of MTRS, nn, is out of range.
The default value, 14000, will be used (CINIT).
```

The value *nn* is the invalid number in the CONFIG file.

Caution

If you use MTRS, you should keep in mind that when you increase the record size, the system will use up more pages of segment 0 to support the larger tape records. (PRIMOS uses segment 0 pages to perform I/O for various devices it supports.) Each device gets exclusive use of its segment 0 pages, and these pages are allocated at coldstart. It is possible that selecting a large record size may cause PRIMOS to run out of segment 0 pages. The result would be that some device that works correctly when MTRS is the default value would fail when a larger value is used, or that PRIMOS will fail at coldstart.

▶ MIRROR

Enables disk mirroring, but does not turn on any mirroring at system startup. See COMDVM and PAGINM. (Use the MIRROR_ON command to start the mirroring of file system partitions.)

▶ NAMLC *number-of-lines*

Specifies the maximum number of directly-connected assignable asynchronous lines.

Parameter	Minimum	Maximum	Default	Recommended
number-of-lines, octal	0	See below.	0	See below.

number-of-lines specifies the maximum number of directly-connected asynchronous lines that may be assigned simultaneously on your system. Assignable NTS lines are not handled by NAMLC. The default is 0. NAMLC + NTUSR + NTSUSR + NTSASL must be less than or equal to 2000_8 (1024 decimal).

The SET_ASYNC command, described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers*, specifies which lines are assignable.

▶ NLBUF *buffers*

Specifies the number of locate buffers to be configured.

Parameter	Minimum	Maximum	Default	Recommended
buffers, octal	10	2000	100	See Ch. 5

buffers is the number of locate buffers for which the system is to be configured. Minimum value is 10_8 (8 decimal) and maximum is 2000_8 (1024 decimal). The default is 100_8 (64 decimal). Each locate buffer occupies 2048 bytes of memory.

▶ **NPUSR** *number*

Sets the number of phantom users.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	4	1700	4	10 or more

number is the number of phantom users for which the system is to be configured. (*number* must be a positive octal integer.) The default value of 4 is also the minimum. If you configure fewer phantoms than this number, PRIMOS prints an error message and uses the default value instead. The maximum is 1700_8 (960 decimal) minus the number of terminal, slave, and remote users (NTUSR, NSLUSR, NRUSR, and NTSUSR).

If you have PRIMENET, you must configure a phantom user for NETMAN. If your system is to be a gateway node, you must also configure a phantom for RT_SERVER. Phantoms are also required for despoolers, the Batch subsystem, the File Transfer Service (FTS), and other purposes.

▶ **NRUSR** *number*

Specifies the number of processes to be reserved for remote logins across the PRIMENET network.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	0	377	0	As needed.

number is the number of remote users for which the system is to be configured. (*number* must be a nonnegative octal integer.) The default is 0. The maximum is 377_8 (255 decimal). NTSUSR+NTUSR+NPUSR+NRUSR+NSLUSR cannot exceed 1700_8 (960 decimal). NRUSR does not apply to terminals connected through modems to asynchronous AMLC or ICS lines.

▶ NSEG *number*

Specifies the number of segments available for all users.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	See below.	20000	1776	See below.

number specifies the number of page maps to be allocated during system initialization. The maximum value is 20000₈ (8192 decimal). The default, 1776₈ (1022 decimal), is often too small. If you set *number* larger than 16000₈, the NVMFS directive may interfere with NSEG. See NVMFS on page 6-22.

NSEG must guarantee at least three segments per configured process. Calculate the minimum value of *number* with the following formula:

$$number \ => \ 3 \ * \ (NTUSR \ + \ NPUSR \ + \ NRUSR \ + \ NSLUSR)$$

If this minimum is not met, the following warning message is displayed during cold start:

WARNING - *m* SEGMENTS MAY NOT BE ENOUGH FOR *n* USERS

where *m* is the number of segments and *n* the number of processes configured. Cold start then continues.

Increase the default value if the condition NO_AVAIL_SEGS$ is frequently signalled on your system.

More page maps may be available than the number of possible user segments (based on available space on the paging partition). If a process cannot get paging space for this reason, the error condition PAGING_DEVICE_FULL$ is signalled.

▶ NSLUSR *number*

Sets the number of slave processes (users).

Parameter	Minimum	Maximum	Default	Recommended
number, octal	0	377	0	As needed.

Each user accessing files on the local system from remote systems requires a slave process for the duration of the access. These slave processes are allocated from the PRIMOS pool of 960 processes.

number is the number of simultaneous remote file accesses the local system wishes to support. If this pool is exhausted when a remote user makes an attach request, the E$NSLA (no NPX slaves available) error code is returned to that user.

The minimum value for *number* is 0, which is the default. The maximum value is 377_8 (255 decimal). NTUSR+NPUSR+NRUSR+NSLUSR+NTSUSR must be less than or equal to 1700_8 (960 decimal) on a 4050, 4150, 6150, 6350, or 6550. On all other systems this limit is 1130_8 (600).

▶ **NTSASL** *number*

Sets the maximum number of NTS assignable lines.

number   The maximum number of NTS assignable lines that can be used simultaneously on your system. The default is 0. The sum of NAMLC + NTSASL + NTUSR + NTSUSR must be less than or equal to 2000_8 (1024).

▶ **NTSUSR** *number*

Specifies maximum number of simultaneous NTS terminal users.

number is the total number NTS terminal users. The default is 0. The sum of NTUSR+NPUSR+NRUSR+NTSUSR+NSLUSR must be less than or equal to 1700_8 (960) on a 4050, 4150, 6150, 6350, or 6550. On all other systems this limit is 1130_8 (600). The sum of NAMLC + NTSASL + NTUSR + NTSUSR must be less than or equal to 2000_8 (1024).

▶ **NTUSR** *number*

Specifies number of terminal users, including the supervisor terminal.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	1	1000	NONE	As needed.

number is the number of directly-connected terminal users for which the system is to be configured. (Do not include network users in this number.) The maximum value of *number* is 1000_8 (512 decimal). NTUSR, which has no default value, must be included in the configuration file.

NTUSR+NPUSR+NRUSR+NSLUSR+NTSUSR must be less than or equal to 1700_8 (960 decimal) on a 4050, 4150, 6150, 6350, or 6550. On all other systems this limit is 1130_8 (600). NTUSR+NAMLC+NTSUSR+NTSASL must also be less than or equal to 2000_8 (1024 decimal). The sum of NAMLC+NTSASL+NTUSR+NTSUSR must be less than or equal to 2000_8 (1024).

▶ NVMFS *number*

Sets the number of VMFA (Virtual Memory File Access) dynamic segments available in virtual address space for the system.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	1	2000	144	400

The maximum value for *number* is 2000_8 (1024 decimal). The default is 144_8 (100 decimal).

The total number of segments available for both NSEG and NVMFS is 20000_8 (8192 decimal). If you specify values for NSEG and NVMFS that total higher than the 20000_8 (8192 decimal) maximum, NVMFS takes precedence over NSEG. For example, if you specify NSEG as 17500_8 (8000 decimal) and specify NVMFS as 454_8 (300 decimal), the system is configured with 454_8 (300 decimal) NVMFS segments available and 17324_8 (7892 decimal) NSEG segments.

VMFA segments are used by EPFs because they can be mapped dynamically. You may want to increase the default number of VMFA segments if users frequently complain that they get messages such as Not enough segments or No space available from process class storage heap.

▶ PAGING *pdev1* [... *pdev8*]

Specifies the paging partitions.

PAGING must be included in the configuration file. A minimum of one paging partition is required, and a total of eight are allowed. *pdev1* through *pdev8* are the physical device numbers of the paging partitions. See the *Operator's Guide to File System Maintenance* for details on construction of physical device numbers.

PRIMOS automatically calculates a strategy to divide paging activity among the available paging partitions. Paging activity is allocated to each paging partition in the ratio of its size to the total size of all the paging partitions. To adjust these ratios, use the PRATIO command, described in the *Operator's Guide to System Commands*.

If all available paging space is used, any attempt by a user process to obtain more memory causes the error condition PAGING_DEVICE_FULL$ to be signalled for that user.

▶ PAGINM *pdev1* [... *pdev8*]

Specifies *pdev1* through *pdev8* as the physical device numbers of the partitions that mirror the corresponding paging partitions specified in the PAGING directive and enables mirroring of file system partitions. This directive turns on paging device mirroring at system startup.

If a *pdev* in the PAGING directive is not to be mirrored, then a 0 must be used in the corresponding position in the PAGINM directive. The PAGINM directive, if used, must follow the PAGING directive in the configuration file. See the *Operator's Guide to File System Maintenance* for further information on mirroring.

▶ **SMLC**

At Rev. 20, SYNC became the preferred synonym for SMLC. For details on the SMLC directives, see the discussions of the SYNC directives, in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

▶ **SYNC**

Here is a summary of the purposes of the four SYNC directives. For extensive and complete information, see the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

SYNC CNTRLR   Associates a device address with a logical controller number and a communications protocol.

SYNC DSC   Specifies data set control information for SYNC, HSSMLC, or MDLC controllers.

SYNC ON   Enables configuration of synchronous communication drivers.

SYNC SYNCnn   Associates a controller's physical line with a logical line number.

▶ **SYSNAM** *name*

Specifies *name* as the name of the system. This directive is required.

The system name uses the PRIMENET nodename syntax: *name* must be from one to six characters, of which the first is a letter, between A and Z. The other characters may be letters or numerals, or any of the seven characters & - $._/#. If PRIMENET or NTS is used, the system name becomes the network nodename.

If you specify SYSNAM with no *name*, PRIMOS prompts the operator to enter a system name at the supervisor terminal. If you omit SYSNAM, or specify a *name* that is too long or contains invalid characters, PRIMOS prints an error message and prompts for the system name.

The erase and kill characters in effect when the operator responds to the system name prompt depend upon the relative placement of the ERASE, KILL, and SYSNAM directives within the configuration file. See the discussion of erase and kill processing under the ERASE directive on page 6-11.

▶ **TYPOUT** $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

Controls displaying of configuration directives at the supervisor terminal.

YES specifies that subsequent directives in the configuration file be displayed at the supervisor terminal as they are processed. Displaying continues until a TYPOUT NO or GO directive is encountered.

NO specifies that commands are not displayed as they are processed. Displaying is suppressed until a TYPOUT YES or GO directive is encountered. TYPOUT NO is the default.

You can put any number of TYPOUT directives into the configuration file to display selected directives.

▶ **UPS** *number*

Controls restart after a power failure.

Parameter	Minimum	Maximum	Default	Recommended
number, octal	0	177777	177777	100

An Uninterruptible Power Supply (UPS) maintains power to the CPU and memory during a power failure and then, on some systems, automatically performs a warm start. If your system has UPS, the value of *number* in the UPS directive determines what action is taken after the warm start.

number has the following values and meanings:

177777   No UPS (default).

0   Produces a warm start followed by a halt. The operator must intervene to bring up the system.

> 0   Number of seconds, in octal, to delay after the warm start before the system comes up. No operator start is required. The number of seconds to delay after a warm start is the amount of time it takes for the disks to come up to the proper number of revolutions per minute. A value of 100_8 (64 decimal seconds) is recommended for a storage module.

If your system does not have UPS, or if it does not automatically warmstart, you should omit this directive.

▶ **WIRMEM**

Prints the size of wired memory (in kilobytes) at the supervisor terminal during cold start.

The size of wired memory changes during the operation of the system. The value displayed by WIRMEM, however, gives some idea of the relative memory cost of the selected configuration. The USAGE command tracks changing wired memory requirements.

APPENDICES

OBSOLETE AND RARELY USED COMMANDS AND DIRECTIVES

This appendix describes two obsolete commands, AMLC, an octal method of configuring asynchronous lines, and LOOK, a debugging tool; five rarely-used configuration directives, FILUNT, MAXPAG, PREPAG, RWLOCK, and VPSD; and eight obsolete configuration directives, ALTDEV, AMLBUF, NTSABF, NTSBUF, REMBUF, PAGDEV, PRATIO, and TPDUMP.

OBSOLETE COMMANDS

▶ AMLC

The AMLC command uses octal bitstrings to configure both terminal and assigned asynchronous lines on the AMLC and ICS controllers. At Rev. 20.2, the AMLC command was replaced by the SET_ASYNC command, a more straightforward way of configuring your asynchronous lines. Although the AMLC command is still supported, its use is no longer recommended.

A complete description of the AMLC command may be found in *System Administrator's Guide, Volume 11: Communication Lines and Controllers.*

▶ LOOK

The LOOK command, which provides access to any segment in the system, is intended as a debugging tool for systems engineers and field analysts. Although System Administrators rarely use the LOOK command, you may use it as a monitoring command.

The LOOK command can be issued only at the supervisor terminal and must be preceded by an OPRPRI 1 command and followed by an OPRPRI 0 command.

The command format of LOOK is as follows:

LOOK [-*usernumber* [*segnumber* [*access* [*mapseg*]]]]

The meanings of the parameters are as follows:

Parameter	Meaning
-*usernumber*	Number of the user owning the segment. The default is user 1. The hyphen must precede the number.
segnumber	Number of the segment to be examined. The default is 6000_8 (the Ring 0 stack segment for the user).
access	Access rights to be granted (as in the SHARE command). The default is 200_8 (read only).
mapseg	Segment of user 1's address space into which the specified segment is to be mapped. The default is 4001_8.

WARNING

Misuse of the LOOK command can destroy system data. The LOOK command can place system integrity at risk if you attempt to examine a segment that does not exist, write to a segment that does exist, or map either shared or stack segments with write permission. The REALLY? prompt is issued for a LOOK command whose request is considered to be risky or dangerous to system integrity. A YES response allows the operation to proceed.

RARELY USED DIRECTIVES

The FILUNT, MAXPAG, PREPAG, RWLOCK, and VPSD configuration directives are not obsolete, but their use should be avoided. These directives are discussed next.

▶ **FILUNT 0** *max-unit*

Sets the maximum number of file units available to each user.

The arguments have the following values and meanings:

0	Prior to Rev 19.4 this parameter specified the maximum number of file units guaranteed to be available to each user. Because the default number of file units available per user is 77772_8 (decimal 32762), it is assumed there will be sufficient file units so that no units need be guaranteed. This parameter must be present, but its value is not used.

max-unit Maximum number of units any one user may have open at one time. The default is 77772_8 (decimal 32762).

If the FILUNT directive is not specified in the configuration file, the default value is used.

You should omit this directive from the configuration file and use the default number of file units.

▶ **MAXPAG** *number-of-pages*

Specifies the number of pages of physical memory (starting from page 0) to be used after cold start.

Parameter	Minimum	Maximum	Default	Recommended
number-of-pages, octal	400	100000	all mem	Omit the directive

number-of-pages is the highest physical page number that will be used, starting from physical address 0. After cold start, PRIMOS uses the first *number-of-pages* of physical memory (that is, from page 0 through *number-of-pages* minus 1). (One page is 2048 decimal bytes of memory; 512 decimal pages is 1 megabyte of memory.)

The value of *number-of-pages* must be between 400_8 (256 decimal) and 100000_8 (32786 decimal). If *number-of-pages* is not specified or if MAXPAG is not in the configuration file, all available memory is used.

The MAXPAG directive is intended primarily for performance testing. It allows a system with a large memory to function as if it had a smaller memory. It may also be used to allow PRIMOS to run successfully on a system that has some bad memory in higher addresses, by making that memory unavailable. Normally, you should not use MAXPAG.

Use the following table as a reference for setting MAXPAG.

Memory (MBytes)	Memory (pages)	MAXPAG argument (octal)	Memory (MBytes)	Memory (pages)	MAXPAG argument (octal)
1	512	1000	14	7168	16000
2	1024	2000	16	8192	20000
3	1536	3000	20	10240	24000
4	2048	4000	24	12288	30000
6	3072	6000	28	14336	34000
8	4096	10000	32	16384	40000
10	5120	12000	64	32768	100000
12	6144	14000			

If your system contains an arrangement of memory boards that produces holes in physical memory (rather than providing a contiguous block of memory), set MAXPAG as if these

holes contained actual memory. For example, specify MAXPAG 4000 (which is 4 MB of memory) if your system has 3.5 MB of memory with a .5 MB hole in the middle.

If you specify a value for MAXPAG that results in the system using less than the total amount of available physical memory, the following message is displayed at the supervisor terminal:

```
System NOT configured with maximum possible memory:
only using mK BYTES, when nK BYTES are available.
```

The message is only a warning, and the MAXPAG directive is obeyed. If you receive this message and you want to use all your available memory, either increase the setting of MAXPAG or remove the directive from the configuration file.

▶ **PREPAG** *pages*

When a page fault occurs and there are no unused memory pages, PRIMOS makes available the three pages that were least recently used by writing them out to disk from memory. To change the default number of pages that are written out, use the PREPAG directive. The argument, *pages*, cannot be less than one or more than the number of pages available for paging. Unless your Prime System Analyst recommends changing the default, omit this directive from the configuration file.

▶ **RWLOCK** *code*

Specifies the system-default setting of the file system read/write lock.

The default value for *code* is 1. The permitted values and their meanings are:

0	1 reader *or* 1 writer, with the writer having exclusive control.
1	N readers *or* 1 writer, with the writer having exclusive control.
3	N readers *and* 1 writer.
5	N readers *and* N writers.

You should omit this directive from the configuration file, because many subsystems and utilities will not work if you change the read/write lock default setting. To change the read/write lock of a file, or a set of files, use the PRIMOS command RWLOCK instead of this directive.

▶ **VPSD**

The VPSD directive wires the Prime assembly language debugger (also known as the kernel debugger) into PRIMOS at system startup. This debugger, formerly used for debugging the

operating system, does not specify any useful system configuration in a production environment.

It is not expected that a typical user environment will need the VPSD directive.

OBSOLETE CONFIGURATION DIRECTIVES

▶ **ALTDEV** *pdev* [*records*]

Specifies the alternate paging partition and, optionally, its size.

At Rev. 21.0 the PAGING directive replaced ALTDEV.

pdev   The physical device number of the paging partition.

records A 16-bit nonzero unsigned integer that indicates the number of records to be used for paging (out of the total number of records available for paging on the partition). The argument is ignored if it is zero, a negative number, or larger than the number of available records. If the argument is ignored (or if *records* is not specified), the entire available space for paging is used.

▶ **AMLBUF** *line* [*in-buff-size* [*out-buff-size* [*dmq-size*]]]

Sets the size of an asynchronous line's I/O buffers.

The AMLBUF directive is obsolete at Rev. 22.0. Use the CAB command, described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

▶ **NTSABF** *line in-buff-size out-buff-size xoff-lag xon-lag*

Sets the size of I/O buffers and flow control thresholds for remote NTS assignable lines connected to the local system.

The NTSABF directive is obsolete at Rev. 22.0, but may be useful for setting flow control thresholds. The CAB command replaces NTSABF; both are described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

▶ **NTSBUF** *line in-buff-size out-buff-size xoff-lag xon-lag*

Sets the size of I/O buffers and flow control thresholds for remote NTS terminal users connected to the local system.

The NTSBUF directive is obsolete at Rev. 22.0, but may be useful for setting flow control thresholds. The CAB command replaces NTSBUF; both are described in the *System Administrator's Guide, Volume II: Communication Lines and Controllers.*

▶ **REMBUF** *in-buff-size out-buff-size*

Sets the sizes of the terminal input and output buffers for remote users.

The REMBUF directive is obsolete at Rev. 22.0. Use the CAB command, described in the *System Administrator's Guide, Volume II:*

Communication Lines and Controllers.

▶ **PAGDEV** *pdev* [*records*]

Specifies the paging partition and, optionally, its size.

At Rev. 21.0 the PAGING directive replaced PAGDEV.

Up to 230,000 records can be used for paging. The values and meanings of the arguments are as follows:

pdev   The physical device number of the paging partition.

records A 16-bit nonzero unsigned integer that indicates the number of records to be used for paging (out of the total number of records available for paging on the partition). The argument is ignored if it is zero, a negative number, or larger than the number of available records. If the argument is ignored (or if *records* is not specified), the entire available space for paging is used.

If all available paging space is used, any attempt by a user process to obtain more memory causes the error condition PAGING_DEVICE_FULL$ to be signalled for that user.

▶ **PRATIO** *n*

Specifies the ratio of use of the alternate paging partition in relation to the primary paging partition.

At Rev. 21.0 the PRATIO operator command replaced the PRATIO directive.

The PRATIO directive is ignored.

▶ **TPDUMP** $\begin{bmatrix} \text{YES} \\ \text{NO} \end{bmatrix}$

The TPDUMP directive is obsolete at Rev. 22.0, and if present is ignored.

TPDUMP previously controlled the ability to take a tape dump before completion of a forced shutdown. The directive is no longer necessary, and a system is always ready for a tape dump following a forced shutdown.

Notes

You no longer need to follow the tape dump with a WARMSTART.

Users of ROAM-based products should not use WARMSTART.

The CRASH_AUDIT facility no longer requires the TPDUMP directive.

INDEX

INDEX

SURVEYS

READER RESPONSE FORM

System Administrator's Guide, Volume I
DOC10131–2LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

☐ *excellent*   ☐ *very good*   ☐ *good*   ☐ *fair*   ☐ *poor*

2. What features of this manual did you find most useful?

_____

_____

_____

_____

_____

_____

3. What faults or errors in this manual gave you problems?

_____

_____

_____

_____

_____

_____

4. How does this manual compare to equivalent manuals produced by other computer companies?

☐ *Much better*   ☐ *Slightly better*   ☐ *About the same*
☐ *Much worse*   ☐ *Slightly worse*   ☐ *Can't judge*

5. Which other companies' manuals have you read?

_____

_____

Name:_____Position:_____

Company:_____

Address:_____

_____

_____Postal Code:_____

READER RESPONSE FORM

System Administrator's Guide, Volume I
DOC10131-2LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

    ☐ *excellent*   ☐ *very good*   ☐ *good*   ☐ *fair*   ☐ *poor*

2. What features of this manual did you find most useful?

    _____
    _____
    _____
    _____
    _____
    _____

3. What faults or errors in this manual gave you problems?

    _____
    _____
    _____
    _____
    _____

4. How does this manual compare to equivalent manuals produced by other computer companies?

    ☐ *Much better*   ☐ *Slightly better*   ☐ *About the same*
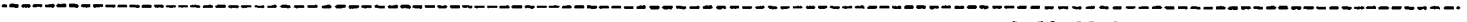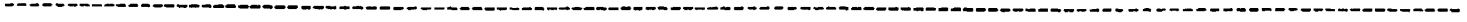   ☐ *Much worse*   ☐ *Slightly worse*   ☐ *Can't judge*

5. Which other companies' manuals have you read?

    _____
    _____

Name:_____Position:_____

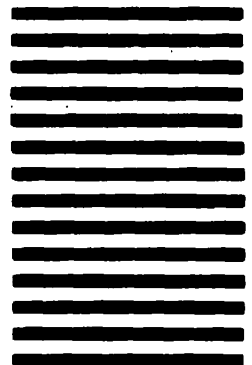Company:_____

Address:_____

_____

_____Postal Code:_____

First Class Permit #531 Natick. Massachusetts 01760

BUSINESS REPLY MAIL

Postage will be paid by:

Prime™

Attention: Technical Publications
Bldg 10
Prime Park, Natick, Ma. 01760

READER RESPONSE FORM

System Administrator's Guide, Volume I
DOC10131-2LA

Your feedback will help us continue to improve the quality, accuracy, and organization of our publications.

1. How do you rate this document for overall usefulness?

   ☐ *excellent*   ☐ *very good*   ☐ *good*   ☐ *fair*   ☐ *poor*

2. What features of this manual did you find most useful?

   _____
   _____
   _____
   _____
   _____
   _____

3. What faults or errors in this manual gave you problems?

   _____
   _____
   _____
   _____
   _____
   _____

4. How does this manual compare to equivalent manuals produced by other computer companies?

   ☐ *Much better*   ☐ *Slightly better*   ☐ *About the same*
   ☐ *Much worse*   ☐ *Slightly worse*   ☐ *Can't judge*

5. Which other companies' manuals have you read?

   _____
   _____

Name:_____Position:_____

Company:_____

Address:_____

_____

_____Postal Code:_____

First Class Permit #531 Natick, Massachusetts 01760

BUSINESS REPLY MAIL

Postage will be paid by:

Prime™

**Attention: Technical Publications
Bldg 10
Prime Park, Natick, Ma. 01760**