RSTS/E System Manager's Guide

Order Number: AA-2762G-TC

This manual describes how to manage a RSTS/E system.

Operating System and Version:	RSTS/E V9.7
Software Version:	RSTS/E V9.7

digital equipment corporation
maynard, massachusetts

July 1989

The information in this document is subject to change without notice and should not be construed as a commitment by Digital Equipment Corporation. Digital Equipment Corporation assumes no responsibility for any errors that may appear in this document.

The software described in this document is furnished under a license and may be used or copied only in accordance with the terms of such license.

No responsibility is assumed for the use or reliability of software on equipment that is not supplied by Digital Equipment Corporation or its affiliated companies.

The postpaid Reader's Comments forms at the end of this document request your critical evaluation to assist in preparing future documentation.

DATATRIEVE
DEC
DECmail
DECmate
DECnet
DECsystem–10
DECsystem–20
DECtape
DECUS
DECwriter
DIBOL

FMS–11
LA
MASSBUS
PDP
P/OS
Professional
Q–Bus
Rainbow
ReGIS
RSTS

RSX
RT
UNIBUS
VAX
VMS
VT
Work Processor

digital™

IBM is a registered trademark of International Business Machines Corporation.

Contents

Preface xvii

Summary of Technical Changes xxi

Chapter 1 Overview of RSTS/E System Management

1.1 Glossary of RSTS/E Terms 1–1

1.2 System Management 1–2

1.3 Disk Organization 1–3
1.3.1 Types of Disks 1–3
1.3.2 File Structures 1–4
1.3.3 Level 0 Disk Structure (Pre-V8.0) 1–5
1.3.4 Level 1 Disk Structure (for V8.0 and V9.0) 1–6
1.3.5 Differences Between Level 0, Level 1.1, and Level 1.2 1–7
1.3.6 Converting Disk Structures (DSKCVT) 1–7
1.3.7 Disk Optimization 1–10

1.4 System Operation Concepts 1–11

1.5 Privilege 1–12

1.6 Multiple Privileges 1–12

1.7 Classes of System Activities 1–12
1.7.1 Account Management Activities 1–12
1.7.2 File Access Activities 1–13

1.8 Multiple Privilege Masks 1–13

Chapter 2 Starting, Halting, and Automatic Recovery of RSTS/E

2.1 Starting up RSTS/E 2–1
2.1.1 Bootstrapping RSTS/E with a Hardware Bootstrap Loader 2–1
2.1.2 Bootstrapping RSTS/E after a System Halt 2–2
2.1.3 Starting Timesharing 2–2

2.2 Halting the RSTS/E System 2–3

2.3 Automatic Recovery and Restart Facilities 2–4
2.3.1 Catastrophic Errors and System Crashes 2–4
2.3.2 Automatic Recovery from Catastrophic Errors 2–5

2.3.3 Automatic Restart Mode Initialization 2–6
2.3.4 Restarting with a Different Monitor 2–6
2.3.5 Restarting with a Different Swap Size 2–6
2.3.6 Power-Fail Hardware Used by RSTS/E 2–7

Chapter 3 The System Startup Command Files

3.1 Creating and Using Command Files 3–1

3.2 System Startup Command File: START.COM 3–2
3.2.1 Mount All Disks 3–6
3.2.2 Assign Systemwide Logical Names 3–7
3.2.3 Start Error Logging 3–7
3.2.4 Enable Caching (Optional) 3–7
3.2.5 Set Default System Characteristics 3–7
3.2.6 Install Run-Time Systems 3–7
3.2.7 Install Libraries 3–8
3.2.8 Set Terminal Characteristics 3–8
3.2.9 Define System Commands 3–8
3.2.10 Set Printer Characteristics 3–8
3.2.11 Start Print/Batch Services 3–9
3.2.12 Start OPSER-Based Spooling Package (Optional) 3–9
3.2.13 Enable LAT (Optional) 3–9
3.2.14 Start Error Analysis 3–9
3.2.15 Enable Logins 3–9
3.2.16 Logout the Initial Startup Job 3–9

Chapter 4 Shutting Down the RSTS/E System

4.1 Performing System Shutdown: SHUTUP 4–1
4.1.1 Running SHUTUP 4–1

4.2 SHUTUP Phases 4–2
4.2.1 Set-Up Dialogue Phase 4–2
4.2.2 Warning Message Phase 4–4
4.2.3 DECnet/E Shutdown Phase (When Necessary) 4–5
4.2.4 Initial Job Killing Phase 4–6
4.2.5 Print/Batch Shutdown Phase (When Necessary) 4–7
4.2.6 OPSER Shutdown Phase (When Necessary) 4–7
4.2.7 EVTLOG Shutdown Phase (When Necessary) 4–8
4.2.8 ERRCPY Shutdown Phase (When Necessary) 4–9
4.2.9 Final Job Killing Phase (When Necessary) 4–9
4.2.10 EMT Logging Shutdown Phase (When Necessary) 4–9
4.2.11 Remove Run-Time Systems and Resident Libraries Phase 4–9
4.2.12 Log File Closing Phase 4–10
4.2.13 Swap File Removal Phase 4–10
4.2.14 Disk Dismount Phase 4–10
4.2.15 Final Shutdown Phase 4–10
4.2.16 SHUTUP Operation Examples 4–10
4.2.16.1 SHUTUP Example: Large RSTS/E System 4–10
4.2.16.2 SHUTUP Example: Small RSTS/E System 4–12
4.2.16.3 SHUTUP Example: DECnet/E Shutdown 4–12
4.2.17 Notes on SHUTUP Operation 4–14

Chapter 5 Managing Quotas

5.1 Disk Quotas 5–1
5.1.1 Disk Quota Operations 5–2
5.1.2 Logged-In Disk Quotas 5–2
5.1.3 Logged-Out Disk Quotas 5–2

5.2 Job Quotas 5–3
5.2.1 Detached-Job Quota 5–3
5.2.2 Job Quotas and Login 5–3

5.3 Send/Receive Quotas 5–4
5.3.1 RIB Quota 5–4
5.3.2 Message Quota 5–4

5.4 Dynamic Region Limits 5–4

Chapter 6 Managing Privileges

6.1 RSTS/E-Defined Privileges 6–1
6.1.1 DATES Privilege 6–3
6.1.2 DEVICE Privilege 6–3
6.1.3 EXQTA Privilege 6–3
6.1.4 GACNT Privilege 6–3
6.1.5 GREAD Privilege 6–3
6.1.6 GWRITE Privilege 6–3
6.1.7 HWCFG Privilege 6–3
6.1.8 HWCTL Privilege 6–4
6.1.9 INSTAL Privilege 6–4
6.1.10 JOBCTL Privilege 6–4
6.1.11 MOUNT Privilege 6–4
6.1.12 PBSCTL Privilege 6–4
6.1.13 RDMEM Privilege 6–4
6.1.14 RDNFS Privilege 6–4
6.1.15 SEND Privilege 6–5
6.1.16 SETPAS Privilege 6–5
6.1.17 SHUTUP privilege 6–5
6.1.18 SWCFG Privilege 6–5
6.1.19 SWCTL Privilege 6–5
6.1.20 SYSIO Privilege 6–5
6.1.21 SYSMOD Privilege 6–6
6.1.22 TMPPRV Privilege 6–6
6.1.23 TUNE Privilege 6–6
6.1.24 WACNT Privilege 6–7
6.1.25 WREAD Privilege 6–7
6.1.26 WRTNFS Privilege 6–7
6.1.27 WWRITE Privilege 6–7

6.2 User-Defined Privileges 6–8

6.3 Multiple Privileges and Jobs 6–8
6.3.1 Privilege Masks 6–8
6.3.2 Job Creation 6–8
6.3.3 Login 6–8
6.3.4 Logout 6–9
6.3.5 Spawned jobs 6–9

6.4 Running Programs Under Multiple Privileges 6–9
6.4.1 Running System Programs 6–9
6.4.2 Temporary Privileges 6–10
6.4.3 Designating a Program as Privileged 6–10
6.4.4 Running a Privileged Program 6–10
6.4.5 Program exit 6–11

Chapter 7 Managing Accounts

7.1 Accounts 7–1
7.1.1 Project-Programmer Number (PPN) 7–2
7.1.2 Account Access Privileges 7–2

7.2 Account Data 7–2
7.2.1 Account Name 7–3
7.2.2 Date/Time/Keyboard of Last Login 7–3
7.2.3 Date/Time of Last Password Change 7–3
7.2.4 Password 7–3
7.2.5 Interactive Flag 7–4
7.2.6 Captive Account Flag 7–4
7.2.7 Network Flag 7–4
7.2.8 Dial-Up Flag 7–4
7.2.9 Expiration Date 7–4
7.2.10 Authorized Privilege Mask 7–4
7.2.11 Creation Date 7–5
7.2.12 UFD Cluster Size 7–5
7.2.13 Quotas 7–5
7.2.14 System Data 7–5

7.3 DCL Command Descriptions 7–5
7.3.1 CREATE/ACCOUNT Command 7–5
7.3.2 DELETE/ACCOUNT Command 7–10
7.3.3 SET ACCOUNT Command 7–13
7.3.4 SET PASSWORD Command 7–16
7.3.5 SHOW ACCOUNT Command 7–18

Chapter 8 Managing Terminals

8.1 Understanding the SET TERMINAL Command 8–1

8.2 Terminal Line Speed Characteristics File: TERSPD.SYS 8–2

8.3 Setting Terminals Other Than Your Own 8–3

8.4 User-Defined Terminal Characteristics File: TERMAC.SYS 8–4

8.5 Automatic Setting of Terminal Characteristics: TERDFL.SYS 8–4

8.6 Setting Permanent Terminal Characteristics 8–5

8.7 Using the /NOBROADCAST and /BROADCAST Command Qualifiers 8–5

8.8 Terminal Servers (LAT) 8–6
8.8.1 Enabling LAT 8–7

8.8.2 Isolation of LAT Problems 8–

8.9 Pseudo Keyboards 8–

8.10 Enabling Forms Management Support (FMS-11) 8–

8.11 DCL Command Descriptions 8–
8.11.1 CREATE/SERVICE/LAT Command 8–
8.11.2 DELETE/SERVICE/LAT Command 8–1
8.11.3 SET NODE/LAT Command 8–1
8.11.4 SET SERVICE/LAT Command 8–1
8.11.5 SET SYSTEM Command 8–1
8.11.6 SET TERMINAL Command 8–1
8.11.7 SHOW COUNTERS/LAT Command 8–2
8.11.8 SHOW NODE/LAT Command 8–2
8.11.9 SHOW SERVICE/LAT Command 8–2
8.11.10 SHOW SESSIONS/LAT Command 8–2
8.11.11 SHOW TERMINAL Command 8–2
8.11.12 SHOW TERMINAL_SERVERS/LAT Command 8–3
8.11.13 START/LAT Command 8–3
8.11.14 STOP/LAT Command 8–3

Chapter 9 Managing Print/Batch Services

9.1 Managing the Print/Batch Services Package 9–
9.1.1 Queue Manager 9–
9.1.1.1 Queue Types 9–
9.1.1.2 Queue Names 9–
9.1.1.3 Default Queues 9–
9.1.1.4 Assigning Servers to Queues 9–
9.1.1.5 Queue Priorities 9–
9.1.1.6 Opening and Closing Queues 9–
9.1.1.7 Starting and Stopping Queues 9–
9.1.1.8 Processing Requests 9–
9.1.1.9 Accessing Entries on a Queue 9–
9.1.1.10 Job Limits for Queues 9–
9.1.2 Print Servers 9–
9.1.2.1 Device Ownership 9–
9.1.2.2 Job Copies and File Copies 9–
9.1.2.3 Header Pages 9–
9.1.2.4 Page Limits 9–1
9.1.2.5 Control Characters and Escape Sequences 9–1
9.1.2.6 Eight-Bit Characters 9–1
9.1.3 Batch Servers 9–1
9.1.3.1 Executing Command Files 9–1
9.1.3.2 CPU and Time Limits 9–1
9.1.4 Managing Forms for PBS 9–1
9.1.4.1 Forms Attributes 9–1
9.1.4.2 The Forms Definition File 9–1
9.1.4.3 Forms Definition Keywords 9–1
9.1.4.4 Changing Forms 9–1

9.2 Multiple Jobs 9–1
9.2.1 The Primary Job 9–1
9.2.2 Secondary Jobs 9–1

9.3 DCL Command Descriptions 9–1
9.3.1 ASSIGN/QUEUE Command 9–2

9.3.2 CLOSE/QUEUE Command 9–21
9.3.3 DEASSIGN/QUEUE Command 9–22
9.3.4 DELETE/QUEUE Command 9–23
9.3.5 DELETE/SERVER Command 9–24
9.3.6 INITIALIZE/QUEUE Command 9–25
9.3.7 INITIALIZE/SERVER Command 9–29
9.3.8 OPEN/QUEUE Command 9–31
9.3.9 SET QUEUE Command 9–32
9.3.10 SET SERVER Command 9–35
9.3.11 SHOW SERVER Command 9–37
9.3.12 START/QUEUE Command 9–39
9.3.13 START/QUEUE/MANAGER Command 9–40
9.3.14 START/SERVER Command 9–42
9.3.15 STOP/QUEUE Command 9–44
9.3.16 STOP/QUEUE/MANAGER Command 9–45
9.3.17 STOP/SERVER Command 9–46

Chapter 10 Managing System Errors

10.1 Use of the Error Logging Programs: ERRINT and ERRCPY 10–1
10.1.1 Error Logging Initialization: ERRINT 10–2
10.1.2 Examples of ERRINT Dialogue 10–3
10.1.3 Error Logging: ERRCPY 10–3

10.2 Displaying Errors: ERRDIS 10–4
10.2.1 Running ERRDIS 10–4
10.2.2 Help Report 10–6
10.2.3 Summary Report 10–6
10.2.4 Bad Block Report 10–7
10.2.5 Adding Bad Blocks to the Bad Block File 10–8
10.2.6 Full Report 10–9
10.2.6.1 User Description in Full Report 10–10
10.2.6.2 Disk Error Detailed Description 10–11
10.2.6.3 MSCP Variations on the Full Report 10–14
10.2.6.4 Nondisk Peripheral Device Error Detailed Description 10–17
10.2.6.5 Nonperipheral Error Detailed Description 10–19
10.2.7 Zero File 10–20

10.3 Analyzing System Crashes: ANALYS 10–21
10.3.1 Running the ANALYS Program 10–21
10.3.2 ANALYS Output 10–23

Chapter 11 Device Management and Caching Control

11.1 Principles of Device Management 11–1

11.2 Optimizing Disk Directory Structure: REORDR 11–2
11.2.1 Why Use REORDR? 11–2
11.2.2 Dialogue Questions and Responses 11–3
11.2.3 Reordering Your Disks 11–4
11.2.4 Error and Processing Messages 11–5
11.2.5 REORDR Example 11–6

11.3 Caching Control 11–7
11.3.1 Size of the Cache 11–8

11.3.2 Sequential and Random Caching Modes 11–8
11.3.3 Random Mode Caching 11–8
11.3.4 Sequential Mode Caching 11–9

11.4 **Caching Guidelines** 11–9

11.5 **Virtual Disks** 11–12

11.6 **DCL Command Descriptions** 11–12
11.6.1 ALLOCATE Command 11–13
11.6.2 CREATE/VIRTUAL_DISK Command 11–14
11.6.3 DEALLOCATE Command 11–16
11.6.4 DELETE/VIRTUAL_DISK Command 11–17
11.6.5 SET CACHE Command 11–18
11.6.6 SET DEVICE Command 11–22
11.6.7 SET PRINTER Command 11–23
11.6.8 SHOW CACHE Command 11–26
11.6.9 SHOW DEVICE Command 11–28
11.6.10 SHOW DISKS Command 11–29
11.6.11 SHOW PRINTER Command 11–30

Chapter 12 Managing System Files and Logical Names

12.1 **Run-Time System Control** 12–2
12.1.1 Adding and Removing Auxiliary Run-Time Systems 12–4
12.1.2 Associating a File with a Run-Time System 12–6

12.2 **Resident Library Control** 12–7
12.2.1 Resident Library DCL Commands 12–7

12.3 **System Logical Names** 12–8
12.3.1 Assigning Logical Names 12–9
12.3.2 Removing Logical Names 12–10
12.3.3 Listing System Logical Names 12–10

12.4 **Defining Concise Command Language Commands** 12–10
12.4.1 Adding a CCL Definition 12–11
12.4.2 Listing Currently Defined CCL Commands 12–12
12.4.3 Removing a CCL Definition 12–13

12.5 **System File Control** 12–13
12.5.1 Adding and Removing Swap Files 12–13
12.5.2 Adding and Removing Overlay and Error Files 12–16
12.5.3 Using the DUMP/SYSTEM Command 12–16

12.6 **DCL Command Descriptions** 12–17
12.6.1 ASSIGN/SYSTEM Command 12–17
12.6.2 DEASSIGN/SYSTEM Command 12–18
12.6.3 DEFINE/COMMAND/SYSTEM Command 12–19
12.6.4 DELETE/COMMAND/SYSTEM Command 12–20
12.6.5 DUMP/SYSTEM Command 12–21
12.6.6 INSTALL/ERROR_FILE Command 12–22
12.6.7 INSTALL/LIBRARY Command 12–23
12.6.8 INSTALL/OVERLAY_FILE Command 12–25
12.6.9 INSTALL/RUNTIME_SYSTEM Command 12–26
12.6.10 INSTALL/SWAP_FILE Command 12–28

12.6.11 REMOVE/ERROR_FILE Command 12–30
12.6.12 REMOVE/LIBRARY Command 12–31
12.6.13 REMOVE/OVERLAY_FILE Command 12–32
12.6.14 REMOVE/RUNTIME_SYSTEM Command 12–33
12.6.15 REMOVE/SWAP_FILE Command 12–34
12.6.16 SET FILE Command 12–35
12.6.17 SHOW COMMAND/SYSTEM Command 12–37
12.6.18 SHOW FILE/SYSTEM Command 12–38
12.6.19 SHOW LIBRARY Command 12–39
12.6.20 SHOW LOGICALS Command 12–40
12.6.21 SHOW LOGICAL/SYSTEM Command 12–41
12.6.22 SHOW RUNTIME_SYSTEM Command 12–42
12.6.23 UNLOAD/LIBRARY Command 12–43
12.6.24 UNLOAD/RUNTIME_SYSTEM Command 12–44

Chapter 13 Job Management

13.1 Controlling the Number of Logged-In Jobs 13–1

13.2 Broadcasting Messages to Terminals 13–3

13.3 Controlling Jobs 13–3

13.4 Removing Jobs 13–4

13.5 Placing a Job on HOLD 13–5

13.6 Setting Job Priority, Run Burst, and Maximum Size 13–5

13.7 Controlling Keyboards and Remote Lines 13–7

13.8 DCL Command Descriptions 13–7
13.8.1 ATTACH Command 13–8
13.8.2 BROADCAST Command 13–9
13.8.3 DETACH Command 13–11
13.8.4 FORCE Command 13–12
13.8.5 HANGUP Command 13–13
13.8.6 LOGIN Command 13–14
13.8.7 REMOVE/JOB Command 13–16
13.8.8 SET JOB Command 13–17
13.8.9 SET SYSTEM Command 13–19
13.8.10 SHOW JOB/PRIVILEGE Command 13–20

Chapter 14 Miscellaneous System Management Functions

14.1 Loading and Unloading Resident SAT 14–1

14.2 Loading and Unloading Monitor Overlays 14–2

14.3 Suspending System Operations 14–2

14.4 Changing System Date and Time 14–3

14.5 Recording System Activities: EMT Logging 14–4
14.5.1 Programming for the EMT Logger 14–4

14.5.2 How EMT Logging Works 14–
14.5.3 Data Returned by EMT Logging 14–
14.5.3.1 Loggable EMTs Enabled or Disabled by SET SYSTEM 14–
14.5.4 EMT Logging and System Security 14–

14.6 Monitoring System Status: SYSTAT 14–

14.7 Dynamic Display of System Status: DISPLY 14–
14.7.1 Running the DISPLY Program 14–
14.7.2 Screen Layout 14–1
14.7.3 Job Status Statistics 14–1
14.7.4 Busy Devices 14–1
14.7.5 Disks 14–1
14.7.6 Message Receivers 14–1
14.7.7 Free Buffers 14–1
14.7.8 Run-Time Systems 14–1
14.7.9 Resident Libraries 14–2
14.7.10 Memory Status 14–2

14.8 Electronic Messaging Systems 14–2

14.9 DCL Command Descriptions 14–2
14.9.1 LOAD/INDEX Command 14–2
14.9.2 LOAD/OVERLAY Command 14–2
14.9.3 SET DATE Command 14–2
14.9.4 SET SYSTEM Command 14–2
14.9.5 SET TIME Command 14–2
14.9.6 SHOW SYSTEM Command 14–3
14.9.7 UNLOAD/INDEX Command 14–3
14.9.8 UNLOAD/OVERLAY Command 14–3

Chapter 15 Managing Disks and Tapes

15.1 Working with Disks and Tapes 15–
15.1.1 Initializing Disks and Tapes 15–
15.1.2 Mounting and Dismounting Tapes and Disks 15–

15.2 DCL Command Descriptions 15–
15.2.1 DISMOUNT Command for Disks 15–
15.2.2 DISMOUNT Command for Tapes 15–
15.2.3 INITIALIZE Command for Disks 15–
15.2.4 INITIALIZE Command for Tapes 15–1
15.2.5 MOUNT Command for Disks 15–1
15.2.6 MOUNT Command for Tapes 15–1

Chapter 16 Backing Up and Restoring Files

16.1 Recovering Your System Disk (RECOVR) 16–

16.2 DCL Command Descriptions 16–
16.2.1 BACKUP Command 16–
16.2.2 RESTORE Command 16–

Appendix A Managing the OPSER Spooling Package

A.1 Overview of Operator Services A–1
A.1.1 OPSER Program Overview A–2
A.1.2 QUEMAN Program Overview A–4
A.1.3 SPOOL Program Overview A–4
A.1.4 BATCH Program Overview A–5
A.1.5 RJ2780 Program Overview A–5
A.1.6 Controlling RESTOR with OPSER A–5
A.1.7 Overview of OPSER Shutdown A–5

A.2 Operator Services Program: OPSER A–6
A.2.1 OPSER Operator Commands A–7
A.2.2 Message Types A–10
A.2.3 Valid Operator and Online Job Lists A–12
A.2.4 Operator INTERRUPT Command A–12
A.2.5 OPSER Startup Procedure A–13
A.2.6 OPSER Action Under Various Startup Conditions A–13

A.3 Queue Manager Program: QUEMAN A–14
A.3.1 QUEMAN Startup Commands and Switches A–16
A.3.2 QUEMAN Interrupt Commands A–17
A.3.3 QUEMAN Startup Procedure A–19
A.3.4 QUEMAN Action Under Various Startup Conditions A–20
A.3.5 QUEMAN Consistency Checking A–20

A.4 Line Printer Spooling Program: SPOOL A–21
A.4.1 SPOOL Startup Options A–22
A.4.2 Line Printer Spooling A–27
A.4.3 Keyboard Spooling A–28
A.4.4 Startup Error Processing A–28
A.4.5 SPOOL Interrupt Commands A–30
A.4.6 SPOOL Startup Examples A–31
A.4.6.1 Line Printer Startup with All Defaults A–31
A.4.6.2 Line Printer Startup with Narrow Width A–31
A.4.6.3 Keyboard Startup on an LA36 . . . . . A–32
A.4.6.4 Keyboard Startup on an LA180 . . . . . A–33
A.4.7 Recovery from Line Printer Errors A–34
A.4.8 Line Printer Output A–35
A.4.9 Error Messages During User Output A–36
A.4.10 Changing and Aligning Forms A–36

A.5 Batch Processor Program: BATCH A–37
A.5.1 BATCH Startup Options A–39
A.5.2 BATCH Interrupt Commands A–40
A.5.3 BATCH Startup Procedure A–41
A.5.4 Operator Action Requests from BATCH A–41

A.6 Operator Communication Program: PLEASE A–42
A.6.1 Running and Terminating PLEASE A–42
A.6.2 OPSER Commands through PLEASE A–43
A.6.3 PLEASE as a CCL Command A–44

A.7 Terminating Operator Services and Spooling A–44
A.7.1 OPSER Shutdown Levels A–44
A.7.2 OPSER Manual Shutdown Procedure A–44

A.8 RESTOR as an OPSER Controlled Program A–4

Appendix B Number Conversion

Appendix C Disk Device Sizes

Appendix D General RSTS/E Messages

D.1 Special Characters Used in Error Messages .... D–

Appendix E SAVRES System Program

E.1 When to Use SAVRES.. E–

E.2 Definitions of SAVRES Terms .... E–

E.3 Running SAVRES .... E–

E.4 SAVRES Switches .... E–

E.5 SAVRES Dialogue... E–
E.5.1 Device Specifications .... E–
E.5.2 Privileges Required .... E–
E.5.3 Checking the Input Volume .... E–
E.5.4 Checking the Output Volume .... E–
E.5.5 Restoring a RSTS/E Disk using the RESTORE Dialogue .... E–
E.5.6 Copying a RSTS/E Disk using the IMAGE Dialogue.............. E–1
E.5.7 IDENTIFY Dialogue E–1
E.5.8 Full Function Command Line E–1

E.6 SAVRES and Booting E–1

E.7 Operator Interface During Processing E–1
E.7.1 Mounting and Dismounting Volumes E–2
E.7.2 Re-Accessing Devices E–2
E.7.3 Aborting SAVRES E–2

E.8 SAVRES Summary Report....................................... E–2
E.8.1 Summary Report Format and Example E–2
E.8.2 Summary Report Run Statistics E–2

E.9 SAVRES Error Handling E–2
E.9.1 General SAVRES Error Messages E–2
E.9.2 Transfer Errors-Fatal and Nonfatal E–2

Appendix F Device Testing Package (DEVTST)

F.1 Introduction to DEVTST F–1

F.2 DEVTST Operating Procedures F–2

F.3 DEVTST Dialogue Errors F–4

F.4 CPEXER F–4
F.4.1 CPEXER Tests F–5
F.4.2 CPEXER Dialogue F–5
F.4.3 CPEXER Errors F–6

F.5 CPUTST F–7
F.5.1 CPUTST Dialogue F–7
F.5.2 CPUTST Errors F–8

F.6 DSKEXR F–8
F.6.1 DSKEXR Dialogue F–9
F.6.2 DSKEXR Errors F–9

F.7 DSKSEK F–10
F.7.1 DSKSEK Dialogue F–10
F.7.2 DSKSEK Errors F–10

F.8 DXEXER F–11
F.8.1 DXEXER Dialogue F–11
F.8.2 DXEXER Errors F–11

F.9 KBEXER F–12
F.9.1 KBEXER Dialogue F–12
F.9.2 KBEXER Errors F–13

F.10 LPEXER F–13
F.10.1 LPEXER Dialogue F–13
F.10.2 LPEXER Errors F–13

F.11 MTEXER F–13
F.11.1 MTEXER Dialogue F–14
F.11.2 MTEXER Errors F–14

Index

Figures

1–1 MFDs and UFDs in Level 0 File Structure 1–5
1–2 MFD, GFDs, and UFDs in the Level 1 File Structure 1–6
9–1 One-to-One Configuration 9–4
9–2 One-to-Many Configuration 9–4
9–3 Many-to-One Configuration 9–5
11–1 Caching Mode Checks 11–10
11–2 Monitor Caching Checks 11–19
13–1 Priority Byte Format 13–5
A–1 System Controlled Programs and Operator Interaction A–3

E–1 Summary Report Format E–22

Tables

2–1 Initialization Option Summary 2–3
6–1 RSTS/E Privileges 6–2
7–1 DCL Commands for Managing Accounts 7–1
8–1 RSTS/E Supported Controller Types 8–2
9–1 DCL Commands for the Print/Batch Services Package 9–1
9–2 PBS Batch Job Exit Status Messages 9–12
10–1 ERRDIS Dialogue Explanation 10–5
10–2 User Description Data 10–11
10–3 Disk Error Detailed Description 10–13
10–4 Nondisk Peripheral Device Format 10–18
10–5 Nonperipheral Error Format 10–20
10–6 ANALYS Program Dialogue 10–22
10–7 System Crash Error Code 10–23
11–1 DCL Commands for Device Management and Caching Control 11–1
11–2 REORDR Dialogue Questions 11–3
11–3 REORDR Message and Response 11–6
12–1 DCL Commands for Managing System Files and Logical Names 12–1
13–1 DCL Commands for Job Management 13–1
14–1 DCL Commands for Miscellaneous System Functions 14–1
14–2 DISPLY Program Switches 14–9
14–3 DISPLY Program Commands 14–11
14–4 STATE Column Attributes 14–15
14–5 Busy Devices Status Abbreviations: WHY Column 14–16
14–6 Disk Status Abbreviations: COMMENT Column 14–17
14–7 Message Receivers Abbreviations 14–18
14–8 Run-Time Systems and Resident Libraries Report Abbreviations 14–19
14–9 Memory Status Report Abbreviations 14–21
15–1 DCL Commands for Disk and Tape Handling 15–1
15–2 Disk Size and Cluster Size 15–8
15–3 The /PUBLIC, /PRIVATE, and /[NO]SHARE Qualifiers for the MOUNT Command . . 15–16
16–1 /INITIALIZE and /OUT_LABEL Interaction During BACKUP 16–8
16–2 BACKUP Quick Reference Table 16–10
16–3 /INITIALIZE and /OUT_LABEL Interaction During RESTORE 16–16
16–4 RESTORE Quick Reference Table 16–18
A–1 OPSER Commands A–8
A–2 OPSER Message and Action Request Contents A–11
A–3 OPSER Online Job List A–12
A–4 QUEMAN Startup Commands A–16
A–5 QUEMAN Startup Switches A–16
A–6 QUEMAN Interrupt Commands A–17
A–7 NEXT Command Error Text A–19
A–8 SPOOL Startup Options A–23
A–9 SPOOL Line Printer Default Values A–27
A–10 SPOOL Keyboard Default Values A–28
A–11 SPOOL Syntax Error Messages A–29

A–12 SPOOL Interrupt Commands A–30
A–13 SPOOL User Requested Output Error Messages A–36
A–14 BATCH Startup Options A–39
A–15 BATCH Interrupt Commands A–40
A–16 BATCH Device Type Designators A–41
A–17 PLEASE Commands to OPSER A–43
A–18 RESTOR Commands through OPSER A–46
B–1 Bit Values in the PDP-11 Word B–1
C–1 Disk Device Sizes C–2
D–1 General RSTS/E Messages D–2
E–1 SAVRES Functions E–3
E–2 SAVRES Switches E–5
E–3 SAVRES Device Specification Switches E–6
E–4 SAVRES Input Volume Error Messages E–7
E–5 RESTORE Dialogue Questions E–10
E–6 IMAGE Dialogue Questions E–14
E–7 IDENTIFY Dialogue Question E–17
E–8 Booting RSTS/E and SAVE Set Volumes E–19
E–9 Summary Report Run Totals E–23
E–10 General SAVRES Error Messages E–25
E–11 SAVRES Nonfatal Transfer Errors E–29
F–1 DEVTST Programs F–2
F–2 RSTS/E Device Designators F–3
F–3 DEVTST Error Messages F–4

Preface

Objectives

This guide explains how to manage a RSTS/E system on a day-to-day basis. To use the features described in this guide, RSTS/E requires you to have certain privileges. Only the RSTS/E system manager and users with the needed privileges should have access to this guide.

Audience

This guide is primarily for use by system managers and operators. You do not need to have a great deal of knowledge of and experience with RSTS/E to manage a RSTS/E system. However, to perform certain system management tasks, you should understand how to use a text editor like EDT and be familiar with the DIGITAL Command Language (DCL).

In addition, because not all customers install their own systems, the manual includes information for DIGITAL field service engineers and software support representatives. This information is clearly indicated.

Document Structure

This manual contains 16 chapters and six appendixes:

Chapter 1 — Contains a glossary of RSTS/E terms and an overview of the RSTS/E disk structures, system management concepts, multiple privileges, account management, and file access.

Chapter 2 — Describes procedures for starting, halting, and restarting RSTS/E.

Chapter 3 — Describes what the system startup command file should include.

Chapter 4 — Describes the phases the RSTS/E system goes through when shutting down.

Chapter 5 — Explains the quotas that limit the amount of various system resources that users can have.

Chapter 6 — Describes the privileges that you can assign to a user when you create an account.

Chapter 7 — Gives background information on managing RSTS/E accounts and describes the DCL commands used to create and maintain the accounts.

Chapter 8 — Gives background information on managing terminals and terminal servers, and describes the relevant DCL commands.

Chapter 9	Gives background information on the Print/Batch Services (PBS) Package and describes the DCL commands for using PBS.
Chapter 10	Describes the available programs and DCL commands for managing system errors.
Chapter 11	Introduces the concepts of device management and caching control on your RSTS/E system and describes the relevant DCL commands.
Chapter 12	Gives background information on RSTS/E system files and logical names and describes the relevant DCL commands.
Chapter 13	Gives background information on RSTS/E jobs and describes the DCL commands for managing them.
Chapter 14	Describes miscellaneous DCL commands.
Chapter 15	Gives background information on working with disks and tapes and describes the DCL commands for managing them.
Chapter 16	Describes the DCL commands for backing up and restoring disk files and accounts.
Appendix A	Describes the OPSER-based spooling package.
Appendix B	Lists the octal and decimal values of each bit in the PDP-11 word.
Appendix C	Lists the cluster size and total size of each disk that RSTS/E supports.
Appendix D	Lists the general RSTS/E messages that you may encounter when using RSTS/E.
Appendix E	Describes the SAVE/RESTORE system program for copying and backing up disks.
Appendix F	Describes the Device Testing Package (DEVTST) used to verify the reliability of RSTS/E supported hardware.

Related Documentation

As system manager, you are responsible for setting certain parameters that affect the monitor and, consequently, the monitor directives. You also need to make decisions related to installing and updating your RSTS/E system. For more information on these topics, refer to:

RSTS/E System Installation and Update Guide

RSTS/E System Directives Manual

RSTS/E Programming Manual

For more information on RSTS/E documentation, consult the *RSTS/E Documentation Directory.*

In addition to the RSTS/E documentation, you should have access to the following manuals:

PDP-11 Peripherals Handbook

DECnet/E System Manager's Guide

Conventions

Symbols and other conventions represent keyboard characters or aid in the presentation of information. This guide uses the following symbols and conventions:

<RET> Represents a carriage-return/line-feed combination. Press the Return key on your keyboard to produce this combination.

<LF> Represents a line-feed character. Press the Line Feed key on your keyboard to produce this character.

<Ctrl/x> Represents the Ctrl key and some other character, represented here by x. For example, enter <Ctrl/U> by holding down the Ctrl key and pressing the key labeled U. RSTS echoes or displays the character at your terminal as ^U.

color In examples, black characters represent data produced by the computer Red characters indicate information that you enter.

[ ] Square brackets show the optional parts of a command in format statements. For example:

DIRECTORY [file-spec[,...]]

In this example, you can include one file specification ([file-spec]), or more than one ([,...]). Square brackets also indicate the choice you hav in using a qualifier. For example:

/[NO]DELETE

This means you can enter either /DELETE or /NODELETE.

| A vertical bar indicates a choice. For example:

/LOWERCASE=[INPUT | OUTPUT]

means you can enter /LOWERCASE=INPUT or /LOWERCASE=OUTP

Summary of Technical Changes

The *RSTS/E System Manager's Guide* for RSTS/E V9.7 contains the following revisions and updates:

- Changing the monitor when restarting the system (Chapter 2)
- Changing the maximum size of the swap file when restarting the system (Chapter 2)
- Running SHUTUP from any local terminal (Chapter 4)
- Running SHUTUP with the automatic restart feature (Chapter 4)
- Managing dynamic region limits (Chapter 5)
- Managing Local Area Transport (LAT) (Chapter 8)
- Managing pseudo keyboards (Chapter 8)
- Enabling the Forms Management System (FMS-11) (Chapter 8)
- Managing the virtual disk (Chapter 11)
- Fine-tuning the EMT logging program (Chapter 14)
- Using electronic messaging systems (Chapter 14)
- Using the revised SAVRES backup utility (Appendix E)

Chapter 1

Overview of RSTS/E System Managemen

RSTS/E runs on a PDP-11 computer. It allows simultaneous, timeshared access to PDP-11 hardware and to RSTS/E software components through either local or remote asynchronous terminals. The *RSTS/E Software Product Description*, which is part of the RSTS/E media kit, describes the hardware and software RSTS/E supports. This guide describes the software a system manager uses to control the operation of a RSTS/E system. The following sections introduce some important RSTS/E concepts.

1.1 Glossary of RSTS/E Terms

To understand how RSTS/E works, you need to know these terms:

CCL (Concise Command Language)
A concise way to run a RSTS/E system program. The CCL syntax lets you run a program without the RUN command and, unlike the RUN command, lets you place the entire command on one line. After the program completes executing, control returns to your job keyboard monitor. The system manager chooses the CCL commands for a particular RSTS/E system. In DCL, CCL's are defined using the DEFINE/COMMAND/SYSTEM command (see Chapter 12).

DCL (Digital Command Language)
A set of commands available on many different DIGITAL systems. These commands perform basic tasks like copying and printing files, and running programs. On RSTS/E, the DCL run-time system manage the DCL command environment. See the *RSTS/E System User's Guid* for more information about DCL on RSTS/E. See the *RSTS/E Quick Reference Guide* for a complete list of all DCL commands available on RSTS/E.

Default Keyboard Monitor
The main keyboard monitor on a RSTS/E system. You enter the defau keyboard monitor (DCL) after you log in.

Job
The unit that RSTS/E uses to keep track of you (and other users) during a terminal session. When you log in, the system creates a job for you and assigns it a job number. The system uses your job numbe to keep track of everything you do from the beginning to the end of your terminal session.

Job Keyboard Monitor
The keyboard monitor that manages a job. The job keyboard monitor the same as the default keyboard monitor, unless you decide to chang it with the SET JOB/KEYBOARD command. After you change the jo keyboard monitor, you remain under its control until you log out or us SET JOB/KEYBOARD again to change it.

Keyboard Monitor

The part of a run-time system which interprets keyboard commands. When working in the Digital Command Language (DCL) environment, for example, you enter commands that the DCL keyboard monitor receives and interprets. Each RSTS/E keyboard monitor has an identifying prompt that it displays to indicate when it expects command input. Common keyboard monitor prompts on RSTS/E are: dollar sign ($) for DCL; "Ready" for BASIC-PLUS; angle bracket (>) for RSX; and period (.) for RT11.

Monitor or Executive

The master control system software that observes, supervises, controls or verifies the operation of a computer system. Also, the collection of routines that controls the operation of user and system programs, schedules operations, allocates resources, and performs I/O.

Operating System

The collection of programs, including a monitor or executive and system programs, that organizes a central processor and peripheral devices into a working unit for the development and execution of application programs.

Program Development

The process of writing, entering, translating, and debugging source programs.

Public Structure

The set of all disks that are public. When you do not include a device name in the file specification, the system by default accesses one of the disks on the public structure. The device name SY: represents the name for all disks in the public structure. Thus, if you do not have any public disks other than the system disk, SY0: and SY: are equivalents. If you have more than the system disk in the public structure, SY: refers to the aggregate of all public disks. SY0: refers to the system disk.

Run-Time System

System software that manages part of the RSTS/E system. For example, the BASIC-PLUS run-time system manages the BASIC-PLUS programming environment. The run-time system controls the interaction between a user or program and the monitor.

System Disk

The disk required by the RSTS/E monitor to start the system and to allow the system to run properly under timesharing. RSTS/E assigns device name SY0: to the system disk.

System Program

Any general-purpose program included in an operating system to perform common functions. CUSP is an acronym for Commonly Used System Program.

1.2 System Management

Management of RSTS/E begins with providing a properly tailored hardware and software configuration, proceeds through initializing the software at system installation time, and continues with the daily functioning of timesharing. To perform these steps efficiently, you should be familiar with timesharing concepts and practices or have a close working relationship with a senior programmer or analyst experienced with timesharing systems.

To manage a system well, you should know the processing requirements of the system and its capabilities and structure. Information in the *RSTS/E System Installation and Update Guide* describes important aspects of hardware and software options and provides memory requirements to assist in configuring a RSTS/E system.

As the system manager, you should be particularly careful about initializing your system for the first time. Because there are so many variables, no prepackaged approach can give you a completely optimized RSTS/E system. The *RSTS/E System Installation and Update Guide* provides detailed information about all the initialization options.

To make sure that important information reaches the right persons within your organization, you should make someone responsible for current documentation of the system. This responsibility includes both locally generated and Digital-supplied procedures and guides. The responsible person should make sure that specified members of the staff receive the most current system information. In too many cases, improper use of resources occurs when responsible individuals d‹ not have the latest information.

1.3 Disk Organization

System managers should understand that the organization of files on disks is important. Only with this knowledge can managers make the best use of all system resources. The following section introduces the concepts of disk and file structures and how to use them to best advantage.

1.3.1 Types of Disks

RSTS/E supports two types of disks as part of its logical disk structure: private and public. The public disk structure consists of a system disk (the one that was booted and contains the RSTS/E monitor) and additional public disk packs or disk cartridges. The system disk must be physically on line and logically mounted whenever the system is running; this way, users can gain access to the system disk during timesharing operations. All other public disks that users may need tc access must be physically on line and logically mounted.

The system disk contains:

- Monitor code
- Initialization code
- The default keyboard monitor (DCL)
- The command files for system startup and crash recovery

Some installations may also use the system disk for storing active user jobs that are temporarily swapped out of memory.

Remaining space on the public disk structure is used for:

- Auxiliary run-time systems
- Optional system files
- System programs
- User programs and data files

You can use disk drives not devoted to the public structure for private disk packs or disk cartridges. Unlike public disks, you can logically and physically dismount and move private disks to other drives during timesharing operations. In addition, private disks make it possible to restrict disk storage to a defined set of users, which can be a distinct advantage when numerous projects are in progress. The file structure of a private disk is the same as that of a public disk.

In addition to private and public disks, both of which any job on the system can share and access, you can logically mount a disk for use by only one job. In this case, only that job can access the disk. When the job terminates (for example, by logging out), the disk is logically dismounted. See "MOUNT Command for Disks" in Chapter 15.

1.3.2 File Structures

The RSTS/E file structures let the monitor access system and user data in an organized way. RSTS/E currently supports three kinds of file structures:

- Level 0.0—is for RSTS/E disks created before V8.0. You can mount disks with this structure as read-only, private disks. You cannot mount them as system or public disks.
- Level 1.1—is the structure created for V8.0. You can mount disks with this structure as read/write, private disks. You cannot mount them as system or public disks.
- Level 1.2—is the structure created for V9.0 and later. You can mount disks with this structure as read/write, system, public, or private disks.

NOTE

> If you are upgrading a single disk RSTS/E system to V9.0, you can mount a Level 0.0 or Level 1.1 disk as the system disk. See the following section "Converting Disk Structures."

The following three sections describe the three kinds of structures. If you have a RSTS/E V9 system with all new disks, you can skip the sections that describe the disk structures before V9. Furthermore, if you have pre-V9 disks, DIGITAL recommends that you use the new DSKCVT disk conversion program to convert them to the latest disk structure (see the section "Converting Disk Structures (DSKCVT)" later in this chapter).

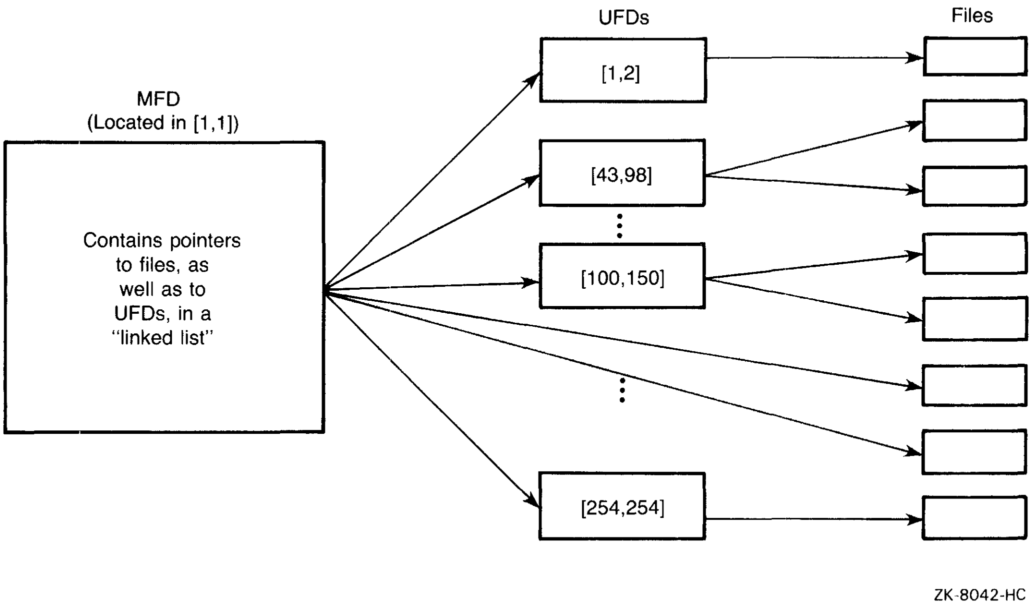
To see which structure applies to disks on your system, type SHOW DISKS as shown. The "Level" column indicates the RSTS/E disk structure level. For example:

```
$ SHOW DISKS <RET>

Disk Structure:
Dsk Open    Size       Free    Clu Err  Name    Level   Comments
DR1   66  131648   29424  22%   4   0  BOWER    1.2    Pub, DLW
DR2    0  242576   33040  13%   8   0  CRONIN   0.0    Pri, R-O, DLW
DR3   33  500352   57744  11%   8   0  WOJTAS   1.1    Pri, DLW
DR4    0  242572   17528   7%   4   0  MODNE    1.2    Pri, DLW
DR5    0  500352   76152  15%   8   0  HOGAN    0.0    Pri, R-O, DLW

$
```

1.3.3 Level 0 Disk Structure (Pre-V8.0)

Pre-V8.0 users gained access to files on a RSTS/E system by a Master File Directory (MFD) and by User File Directories (UFDs). Figure 1–1 shows these structures.

Figure 1–1: MFDs and UFDs in Level 0 File Structure



As Figure 1–1 shows, each disk initialized for use on a RSTS/E system contains an MFD, located in [1,1]. The system uses the MFD to catalog other accounts on the disk. The MFD on the system disk lists the accounts that can be used to log in to the system. On a private disk, the MFD contains entries of accounts that can create files on that disk. Any user can access any file on any disk if the protection code of the file permits. Only users whose accounts are entered in the MFD on a private disk can create files on that disk.

The system creates one UFD for each user account on a disk when the manager sets up the account (or, optionally, when a user creates the first file for the account). The UFD catalogs all program and data files under an account and maintains accounting and access information for these files. The UFD contains all retrieval information for the files because each file is pure data and has no linkage or structural information.

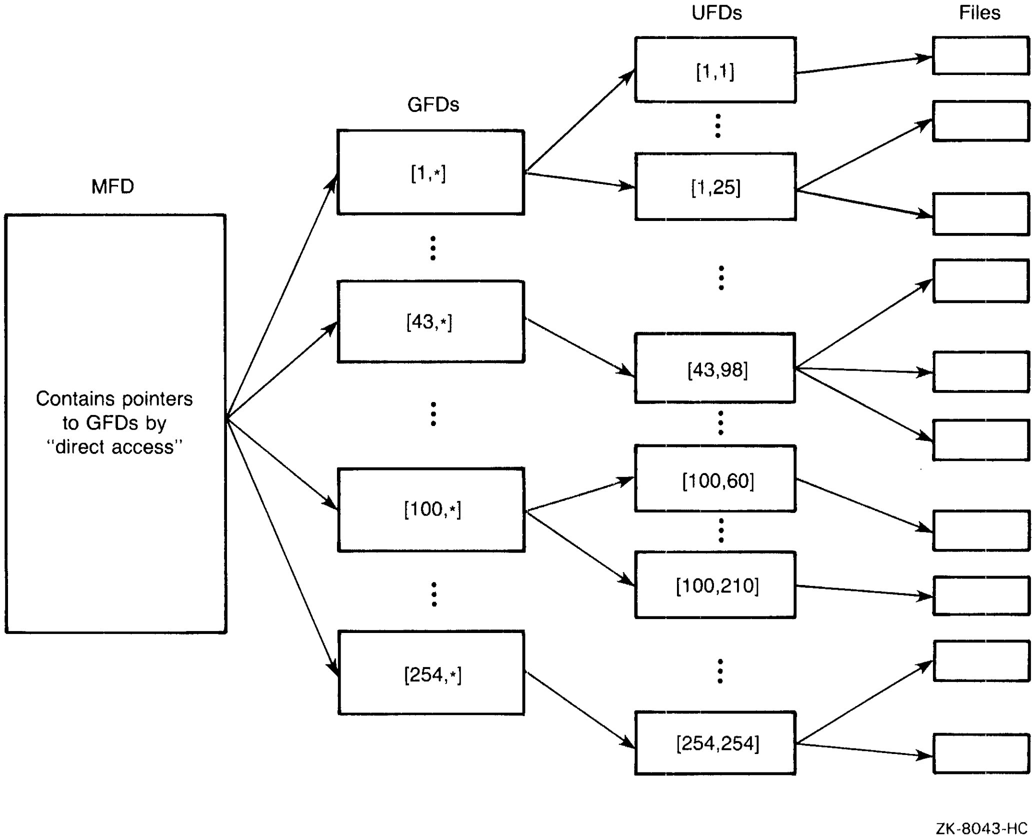
1.3.4 Level 1 Disk Structure (for V8.0 and V9.0)

In V8.0 and V9.0, users access files on a RSTS/E system through the use of three structures:

- A Master File Directory (MFD)
- A Group File Directory (GFD)
- A User File Directory (UFD)

Figure 1–2 shows how these structures are related.

Figure 1–2: MFD, GFDs, and UFDs in the Level 1 File Structure



As Figure 1–2 shows, the MFD contains pointers to the Group File Directories (GFDs). The GFDs point to UFDs within each group. For example, a pointer in the MFD could access the GFD for files in [43,*] directories, from which it could access a specific UFD, such as [43,98].

1.3.5 Differences Between Level 0, Level 1.1, and Level 1.2

The new disk structure (Level 1.2) differs from the previous disk structures (Leve 0 and Level 1.1) in the following ways:

- The account data on Level 1.2 contains attribute blockettes for the new or expanded attributes such as quotas, multiple privileges, and long passwords.
- The MFD on Level 1.1 and Level 1.2 is not associated with an account number.

  On Level 0, the MFD was located in [1,1], and had to be at Block #1. Now, Block #1 points to the location of the MFD, allowing the MFD to be placed anywhere on the disk.
- The MFD and GFDs on Level 1.1 and Level 1.2 do not store files.

  On Level 0, the MFD could store files as well as point to accounts.
- GFDs on Level 1.1 and Level 1.2 allow more direct access to accounts.

  On Level 0, the MFD contained a linked list of pointers to all the UFDs. UFDs were listed in the order in which they were created; the MFD was searched in sequence until it located a UFD.

  Now, the MFD contains direct pointers to GFDs, which in turn point directly to any UFD. (However, the structure of the UFDs remains the same as in Level 0.) Thus, DCL commands that display accounts (for example, SHOW ACCOUNT) list the accounts in ascending order, rather than in order of creation.
- You can create up to 65,025 accounts with the Level 1.1 or Level 1.2 structure

  With the Level 0 structure only 1735 accounts could be created and referenced on a disk. Now, you can create accounts for 255 groups, each with 256 users. Only the disk size restricts the number of accounts.

The users on your system should not see the differences between the three structures. Level 1 takes up a little more disk space. There are more pointers, because each group is handled separately, but access to files is faster and the system can have more accounts. The linked list structure of the MFD is replaced by direct access to GFDs.

1.3.6 Converting Disk Structures (DSKCVT)

The DSKCVT program performs the following functions:

- Converts Level 0.0 disks to Level 1.2
- Converts Level 1.1 disks to Level 1.2
- Converts accounts with LOOKUP passwords to NOLOOKUP (optional)

DSKCVT converts disks directly; you need no second disk to perform the conversion. DIGITAL recommends that you first back up the data on the disks to be converted, so that any hardware or software errors that may occur during conversion will not cause loss of data.

DSKCVT requires some work space if the pack being converted is Level 0.0 format. Make sure there are 32 blocks times the number of different group numbers used on the pack. For example, if there are ten group numbers, the pack should have 320 blocks of free space for conversion.

DSKCVT accepts either logically mounted or logically dismounted packs. However, the pack should not be in use when you run DSKCVT. To make sure no one is using the pack, DSKCVT dismounts and remounts it at the start of the conversion.

In converting Level 1.1 disks to Level 1.2, DSKCVT adds several new atrtibute blocks to each account. Since some of the new attributes are only relevant to user accounts, DSKCVT asks:

```
Convert accounts to NOUSER <no>?
```

If you answer yes to this question (the default is no), DSKCVT changes all accounts to non-user accounts. Non-user accounts have no passwords or privileges and can only serve to store files.

CAUTION

> If you intend to move the accounts (via BACKUP and RESTORE) to the system disk from a disk you are converting with DSKCVT, then *do not* convert the accounts to NOUSER. Otherwise, you will not be able to log into the accounts or create jobs in them.

DSKCVT converts accounts to Level 1.2 by adding the following new attribute blocks to each account:

- Disk quota/usage block
- Privilege block
- Nondisk block

In creating the privilege attribute block, DSKCVT uses the following rules:

- For account [1,1], it assigns all privileges except EXQTA
- For all other [1,*] accounts, it assigns all privileges except SYSMOD and EXQTA
- For accounts not in the [1,*] group, it assigns no privileges

This scheme assigns privileges that are compatible with previous versions of RSTS/E. Once the conversion is complete, you can modify privileges as necessary using the SET ACCOUNT/PRIVILEGE command. For example, if you want all users to be able to change their password, type:

```
$ SET ACCOUNT/PRIVILEGE=SETPAS [*,*]
```

DSKCVT sets the quota values as follows:

- Logged-Out Disk Quota: Quotas of 0 (unlimited in V8.0 and earlier) are changed to -1 (unlimited in V9.0 and later). Other values are retained.
- Logged-In Disk Quota: Set to unlimited.
- Job Quota: Set to unlimited.
- Detached Job Quota: Set to unlimited for accounts [1,*], zero for others.
- RIB Quota: Set to unlimited for accounts [1,*], 3 for others.
- Message Quota: Set to unlimited for accounts [1,*], 12 for others.

These values are compatible with V8.0. Again, you can use the SET ACCOUNT command to change them as needed.

The accound flags are set as follows:

- NOEXPIRE (If the account's password was ?????? then the account is marked EXPIRED.)
- NOCAPTIVE
- INTERACTIVE
- DIALUP
- NETWORK
- PASSWORD_PROMPT
- LOOKUP or NOLOOKUP (according to what you specified)

NOTE

> Accounts with any ?s in their passwords convert properly but cannot be logged into, since ? is an invalid character in V9.0 and later. This is compatible with the meaning of ? in passwords in previous versions.

When converting a disk, you can decide whether to convert all account passwords to their hashed form (NOLOOKUP) or leave them in their readable form (LOOKUP). If your site still uses applications that must look up passwords, do not hash passwords on the converted disk. (When you no longer need LOOKUP passwords, you can use DSKCVT to convert all account passwords to NOLOOKUP on an existing Level 1.2 disk.)

If you convert all accounts to NOUSER accounts, then DSKCVT will not create a password, privilege, or non-disk quota block for all accounts on the disk.

You need **all** privileges to run DSKCVT. DSKCVT displays an error and aborts if you do not have all privileges currently enabled. There are two ways to obtain all privileges in order to use DSKCVT:

- Log into an account that has all privileges. Such an account has all the power and capabilities of the [1,1] account in V8.0, including the ability to poke memory (SYSMOD privilege).
- During system startup, answer no to the Proceed with system startup? prompt. You will return to the DCL prompt and will be running in a job that has all privileges. You can then run the DSKCVT program to convert disks as necessary. Once you have finished using DSKCVT, you can resume system startup by typing the command:

  ```
  $ @[0,1]START
  ```

The second method shows that you do not require any account on your system that has all privileges. Whenever you need to perform an operation that requires a privilege not assigned to any account, you can do so by interrupting the initial system startup procedure.

The following is an example of the DSKCVT program:

```
$ RUN $DSKCVT
DSKCVT V9.3-03   RSTS V9.6-11

Disk to convert? DU2:

** CURRENT DISK ATTRIBUTES **

    Disk:     _DU2:
   Label:     WORK
   Level:     1.1
```

```
Convert accounts to NOUSER <NO>?
Convert passwords to NOLOOKUP <YES>?
Display accounts <YES>?
Proceed (YES or NO)? Y

Converting disk _DU2: to level 1.2 ...
Account [  0,1  ] converted
Account [  1,1  ] converted
Account [  1,2  ] converted
            .
            .
            .

Converting disk _DU2: passwords to NOLOOKUP ...
Account [  0,1  ] is a NOUSER account
Account [  1,1  ] is a NOUSER account
Account [  1,2  ] is a NOUSER account
            .
            .
            .

Disk _DU2: conversion completed on 19-Apr-90 at 11:04 AM
```

In the example, DU2: is the disk pack to convert to Level 1.2. When you type DU2: in response to the prompt, DSKCVT displays the pack ID DU2:WORK. This lets you verify that you are converting the right pack. If the pack was already converted, DSKCVT displays an error message and then repeats the Disk to convert: prompt. Otherwise, DSKCVT goes on with the dialogue.

When you answer Y to the Proceed (YES or NO)? prompt, DSKCVT begins the conversion. DSKCVT processes the accounts in ascending order. When the pack is converted, DSKCVT dismounts it and updates the pack label. The pack is now in Level 1.2 format. However, the storage allocation table (SAT) does not yet reflect the free disk space on the pack. Therefore, the system marks the pack "dirty" until DSKCVT updates the SAT. Then the system updates the pack label to show that the pack is now rebuilt. If the pack was mounted when DSKCVT started, the system mounts it once again, with the pack ID as its logical name.

DSKCVT will not lose data even if the program aborts because of lack of work space, a system crash, or a disk hardware problem. If there is a problem during the conversion, you need only correct the problem and rerun the conversion. In such a case, you do not need to restore the pack from a backup pack.

If the conversion proceeded far enough that DSKCVT was able to convert the pack, you only need to mount it. Otherwise, DSKCVT restarts the conversion from the beginning, after deleting the work files left over from the first attempt at conversion.

When you convert a Level 0.0 (pre-V8.0) disk, DSKCVT makes two passes. In the first pass, DSKCVT converts the Level 0.0 pack to Level 1.1. The second pass converts the Level 1.1 pack to Level 1.2.

1.3.7 Disk Optimization

When a user tries to access a file on the public structure, the system searches the directories on all disks that are part of the public structure. The search verifies whether the file exists. You can avoid the overhead of searching more than one disk by placing frequently accessed files on a private disk. This technique is valid for systems with a number of disks in the public structure, as well as for a system whose public structure consists of only one disk. The ability to place and preextend directories can also help you optimize disks (see Chapter 7).

It is sometimes an advantage to dedicate an entire private disk to a single large production file. Such organization ensures an efficient directory structure and minimizes overhead to access file data. When more than one file is on the same private disk, it is best to dedicate a whole account to each production file. The system then spends less time searching the directory for file information.

In general, you should keep volatile files and stable files in separate accounts to avoid fragmentation. Where file entries in the directory are spread across the face of the disk rather than being confined to a few sequential blocks, fragmentation requires the disk head to move more than is optimally efficient. Preextending directories can ensure that the directory is in sequential blocks (see Chapter 7). REORDR can restructure the directories on your system to improve disk access time (see Chapter 11). You can also optimize disk usage by separating frequently accessed files from those you use less often.

In an environment where distinct data files must be accessed by the same program, the optimal organization is to keep each file on a different private disk. If a program must access more than one file on the same disk, overhead is increased because of disk-head movement. A large percentage of time is spent, therefore, in moving the head back and forth. However, if each file referenced by the program exists on a distinct private disk, head movement is not required whenever a program references another file. Head movement is restricted to locating the data itself. Positioning of directories can also help reduce head movement (see Chapter 7).

1.4 System Operation Concepts

Immediately after logging in to the system, a user's terminal is under the control of DCL (the system's default keyboard monitor). The terminal is said to be at the system command level because you can type a system command and the keyboard monitor processes it accordingly. The terminal returns to the system command level when a command or program completes executing, or when you enter a Ctrl/C at the terminal. When the terminal is at the system command level, the keyboard monitor examines each ASCII text line entered and determines whether that line is a system command. System commands are executed upon entry as described in the *RSTS/E System User's Guide.*

The user job area is initialized at log-in time and set to a size of at least 1K words (K = 1024). The job area can grow in increments of 1K words to a maximum size set by the system manager. The maximum size for any job image, including memory used by the current run-time system of the job, is 64K words. This is the maximum size that the PDP-11 Instruction and Data (I&D) architecture (PDP-11/44,45,50,53,55,70,73,83,84) allows. For a PDP-11 without I&D architecture (PDP-11/23+,24,34,35,40,60) the maximum size is 32K words.

Under the RSTS/E system, jobs run one at a time. A job runs until it enters an Input/Output (I/O) wait state or until it exhausts the time quantum assigned to it either by the system or the system manager. When the current job stops running, the scheduler runs the next job that is ready. Meanwhile, the interrupt-driven I/O device handlers are processing requested data transfers. After completing a transfer, the scheduler marks the job that requested the transfer as ready to run again and starts it from the point at which execution stopped.

RSTS/E tries to keep as many jobs in memory as possible. When a job requires more memory than is available, the system may move some jobs out of memory to temporary storage in one of up to four areas known as swap files. This operation is called swapping. By convention, these files are named SWAP0.SYS, SWAP1.SYS, SWAP.SYS, and SWAP3.SYS. When memory becomes available for a job stored in a swap file, it is swapped back into memory. Jobs waiting for keyboard input and jobs waiting for device I/O completion are most likely stored in the swap files, while jobs currently running or involved in disk or magnetic tape data transfers are stored in memory.

As the system processes each job, it maintains accounting information in memory for that job. When the job is logged off the system, the information is used to update the accounting information stored on the system disk for that account.

1.5 Privilege

You must have a way to prevent general access to activities that can damage the system. The multiple privileges feature gives you finite control over access to activities. You can limit user access to just those activities suitable to each job. Multiple privileges gives you a tool to enhance both system performance and security.

1.6 Multiple Privileges

The multiple privileges feature groups similar system functions into sets and defines a privilege to control access to each set of functions. A group of 35 privileges govern the entire set of RSTS/E system functions. The privileges given to an account determine the range of functions available to the user. Some privileges apply to very specific functions, while others control functions within broader classes of system use (see Chapter 6).

1.7 Classes of System Activities

The following sections divide some computer activities into general classes of system functions and discuss the privileges that control them. They describe:

- Account Management Activities
- File Access Activities

1.7.1 Account Management Activities

One general class of system activity is account management. A user accesses a computer through an account. The individual account is a member of a group, which contains all other accounts with the same project number. The group, in turn, is a subset of a world, which contains all accounts on the system. Account management activities include creating and deleting accounts, as well as changing passwords, disk quotas, and expiration dates. The following privileges control account management:

GACNT	Group Account Management—Grants account management privileges within the user's group.
WACNT	World Account Management—Grants account management privileges for all accounts.
SETPAS	Set Password—Allows changing one's own password.

1.7.2 File Access Activities

Another general class of system activity is file access. Users routinely access files. Some files are created by the user and reside in the individual's account. Other files reside in the accounts of other users or in system accounts. File acces activities include creating, deleting, renaming, reading, writing, and executing files.

Both the protection code of the file and the privileges granted to the user can affect whether the system grants or denies file access.

In the case where no privileges are granted to a user, protection codes control th types of operations that a user can perform on a file. The SET PROTECTION command (or the /PROTECTION qualifier in the RSTS/E file specification) passe a value to the system that sets bits in the protection code byte. Each bit specifie that the activity named by that bit is prohibited or allowed.

Certain privileges also govern file access activities. Some privileges override protection codes completely. The following privileges grant a user the right to perform certain file access activities, regardless of protection codes:

GREAD — Group Read—Read any file within the group. Also, execute any program within the group, if the program's executable bit is set.

WREAD — World Read—Read any file on the system. Also, execute any program on the system, if the program's executable bit is set.

GWRITE — Group Write—Create, delete, extend, modify, or rename any file within the group.

WWRITE — World Write—Create, delete, extend, modify, or rename any file on the system.

NOTE

In addition to WWRITE, you need SYSIO privilege to create, delete, extend, modify, or rename any file in group zero ([0,*]).

1.8 Multiple Privilege Masks

You assign a certain set of privileges to each account. The system stores this set of privileges in privilege masks. A privilege mask is a set of flag bits with one bit corresponding to each privilege.

For each active job, RSTS/E keeps 3 masks:

- Authorized mask—The set of privileges that you give to the account. Users can use the SHOW ACCOUNT/FULL command to list the set of privileges available to their own account.
- Current mask—The set of privileges now in effect for the job. The system always references this mask when it performs a privilege check. Users can raise or lower their privileges (up to their authorized limit) with the SET JOB/PRIVILEGE command.
- Saved mask—The saved record of the current privileges when a job gains temporary privileges (see the section "Temporary Privileges" in Chapter 6).

When a user attempts to perform an activity that is restricted by one or more privileges, the system performs a privilege check. It examines the current mask to determine if the requesting job has all the privileges required to perform the activity. If the requesting job has insufficient privilege to perform the activity, the system returns one of the following errors:

- ?Protection violation
- ?Illegal SYS( ) usage
- ?XXXX privilege required

Chapter 2

Starting, Halting, and Automatic Recovery o RSTS/E

This chapter describes how to start and halt a RSTS/E system and how the system recovers automatically from catastrophic system errors. This chapter also describes how to create the startup conditions for a RSTS/E system. See the *RSTS/E System Installation and Update Guide* for a description of all other initialization code options and their functions.

2.1 Starting up RSTS/E

To start a RSTS/E system, you must first load into memory a program containing the initialization code (INIT.SYS) and then answer YES to the Start timesharing <YES> prompt.

INIT.SYS is a large, stand-alone program that, once you transfer it from the RSTS/E distribution kit, exists on the system disk. INIT.SYS can provide a number of important system functions, such as:

- Creating the RSTS/E file structure
- Creating system files
- Installing a RSTS/E monitor
- Setting system defaults
- Creating startup conditions for RSTS/E

INIT.SYS does some of these automatically; however, they can also be done through the Option: prompt.

Before you can use the initialization code to start your RSTS/E system, you must move the code into memory. This is called bootstrapping. The following sections describe how to transfer the intialization code into memory.

2.1.1 Bootstrapping RSTS/E with a Hardware Bootstrap Loader

The procedures for bootstrapping RSTS/E depend on the type of hardware bootstrap loader and the type of disk you use as the system device. You should make sure:

- The system disk is physically mounted on a disk unit.
- The drive on which the system disk is mounted has the READY light on and is in the WRITE ENABLE condition.

- All required units are running and READY.
- The console terminal is on line.

Read the *RSTS/E System Installation and Update Guide* for the proper instructions on using the hardware bootstrap on your system.

After you use the bootstrap to load the initialization code, INIT.SYS marks the successful completion of the bootstrap procedure by printing an identification line that includes the RSTS version number, the installation name, the disk unit bootstrapped, and the INIT version number, followed by the Current date prompt. Enter the date and press Return. INIT displays the Current time prompt. Enter the time and press Return. INIT displays the Start timesharing prompt. For example:

```
RSTS V9.7 installation-name (DM0) INIT V9.7

Current date <DD-MMM-YY>? 15-Jan-90<RET>
Current time <HH:MM AM/PM>? 8:15 AM<RET>

Start timesharing? <Yes>
```

If the program does not print this information, make sure the console terminal is on line and try the operation again. If the system prints an error message on the console terminal, see the *RSTS/E System Installation and Update Guide* for recovery procedures.

If you want to enable the automatic restart facility, set the CPU switch register so bit 0 is ON. The automatic facility remains enabled as long as the CPU switch register remains set in this way. If your CPU does not have a switch register, the automatic restart facility is always enabled.

2.1.2 Bootstrapping RSTS/E after a System Halt

When a RSTS/E system halts as a result of a catastrophic error, RSTS/E displays the halt address in the address lights. For central processors that do not have console lights, for example the 11/23+, the 11/44, and CPUs with Remote Diagnostic Consoles (RDCs), the system prints the address on the console terminal. It is essential that you record this address. The exact procedure for examining memory locations depends on the type of processor and front panel of your system. Your Digital Software Specialist can describe the procedure for your system. Once you record this information, bootstrap your system again using the procedures described in the previous section. See the section "Automatic Recovery and Restart Facilities" for a description of catastrophic errors, system crashes, and crash dumps.

2.1.3 Starting Timesharing

After you bootstrap the system disk, RSTS/E loads the initialization code (INIT.SYS) into memory. INIT then prints an identification line followed by the Start timesharing? <Yes> prompt on the console terminal. You can answer in one of four ways:

- Type YES or press Return to start the monitor named in the INIT logo.
- Type START and press Return to start the monitor named in the INIT logo and display a list of disabled devices.

- Type START monitor-name and press Return to start another monitor in [0,1] and display a list of disabled devices.
- Type NO to have INIT return to the Option prompt.

  The program is ready to accept one of the options that Table 2–1 summarizes. See the *RSTS/E System Installation and Update Guide* for a complete summary of these options.

Table 2–1: Initialization Option Summary

Option	Abbreviation	Meaning
BOOT dev	BO	Boots another device.
DATE	DA	Resets the current date and time. Also, establishes default date and time formats.
DEFAULT	DE	Establishes or changes a monitor's startup defaults such as memory usage.
DSKINT	DS	Initializes and optionally formats a disk. It also checks for bad blocks.
HARDWR	HA	Specifies device controller characteristics.
HELP	HE	Prints the Help text.
IDENT	ID	Prints INIT.SYS creation date and a list of patches for INIT.SYS and the installed monitor.
PATCH	PA	Modifies RSTS/E system code to correct program errors and to add new program features.
REFRESH	RE	Creates and positions system swap files, adds bad blocks to the bad block file, and positions optional system files on the system disk or any auxiliary disk.
SAVRES	SA	Reconstructs (RESTORE), or makes an image copy (IMAGE) of a RSTS/E disk.
START sil	ST sil	Installs the specified Save Image Library (SIL), assigns and checks defaults, and starts the RSTS/E system. If you do not enter a name for the SIL, INIT prompts you for one.

2.2 Halting the RSTS/E System

A system crash or setting the CPU console HALT/ENABLE switch to its HALT position causes a RSTS/E system to halt. For the PDP-11/45, 50, 55, and 70, RSTS/E considers the processor to be in a halt state if both RUN and PAUSE lights are off. For all other PDP-11 processors, RSTS/E considers the processor to be in a halt state if the RUN light alone is off.

The SHUTUP program shuts down the RSTS/E system. During this operation, SHUTUP makes sure all files are properly closed and that system accounting information is accurately updated. If you want to turn the power off, run SHUTUP, wait for INIT's Start timesharing? prompt, and press the HALT/ENABLE switch. See the section "Performing System Shutdown: SHUTUP" in Chapter 4 for a complete description of the SHUTUP program.

Digital recommends that you do not halt RSTS/E during timesharing by moving the HALT/ENABLE switch on the CPU console to its HALT position. If you do this during system shutdown, SHUTUP may not have completed clean-up operations; consequently, file data can become corrupted and accounting information may be lost. The only way to recover from such a halt and to salvage file information is to do the following:

1. Raise the HALT/ENABLE switch back to its ENABLE position before you take any other action
2. Press the CONT switch to return the RSTS/E system to the state it was in before the halt

2.3 Automatic Recovery and Restart Facilities

This section describes how catastrophic errors occur and how your system can automatically recover from them. This section also points out the need to include commands in the START.COM file to perform critical system operations when the system restarts in automatic mode.

2.3.1 Catastrophic Errors and System Crashes

Catastrophic errors and system crashes result from unexpected error traps. (For information on error traps, see the related processor handbook.) The processor can cause unexpected error traps when it:

- Refers to a nonexistent or nonresponding memory address (for example, bus time-out trap)
- Refers to an odd address with an instruction that requires a word address
- Attempts to execute a reserved or nonexistent instruction

Also, if the monitor detects a condition which it knows should not normally occur, the monitor can force a software crash. Catastrophic errors and system crashes may occur as a result of three types of problems:

- Privileged job programming errors

  RSTS/E system software is designed to protect itself against programming errors occurring in nonprivileged jobs. When it detects such an error, the system aborts execution of the user request and returns a corresponding error message to the user's program.

  RSTS/E software is vulnerable to some types of errors caused by jobs with certain privileges. The privileges that require special caution are SYSIO, SYSMOD, and WRTNFS. Even though most errors do not cause the system to crash, you should take special care when programming with privileged system function calls such as Poke memory SYS call (SYS -6). See the *RSTS/E Programming Manual.*

- Hardware malfunctions

  Hardware malfunctions can cause the system to crash. If system crashes or catastrophic errors occur that you cannot explain (particularly on systems that have traditionally been functioning well), it is likely that a hardware problem has occurred. You can diagnose hardware problems by examining the output logged by the programs in the RSTS/E System Error Package. See Chapter 10 for information on error logging.

- System software malfunctions

  Although every attempt has been made to detect and eliminate system software errors, it is possible that RSTS/E software can crash unexpectedly. If a problem of this type is discovered (it should be reproducible in a defined environment and under defined conditions), contact a Digital Software Specialist. As problems of this type become known, Digital reports them in the "RSTS/E Software Dispatch" and corrects them as soon as possible.

2.3.2 Automatic Recovery from Catastrophic Errors

The system takes action in one of two ways when an unexpected error trap occurs:

- A catastrophic error that affects only one user
- A system crash for which some software or hardware problem may be responsible

For a catastrophic error, the system determines which user was responsible for the error. It flags the user's job with a special code; this causes the system to completely reinitialize that user's job area when it is the user's turn to run. The system prints an error message on the user's terminal, followed by the text: ??Program lost-sorry. The system resumes normal time-sharing operations.

NOTE

Some run-time systems may issue other error messages.

When the system detects a condition from which it cannot recover, it does the following:

1. Writes an image of read/write memory and tables to the CRASH.SYS file.
2. Bootstraps the system disk.
3. Loads the initialization.code into memory.
4. Rebuilds the system disk.
5. Executes an automatic restart. (This occurs if the CPU switch register has bit 0 set and if at least one minute has passed since startup.)

If the CPU register does not have bit 0 set or if one minute has not passed since startup, the system does not perform an automatic restart. The system will only reboot; it will not restart. In other words, the automatic restart operation restarts timesharing using commands in the START.COM file. In contrast, a reboot merely loads the initialization code (INIT.SYS) into memory. INIT.SYS then prints its Start timesharing? <Yes> prompt, to which you can answer by typing any of the options in Table 2–1.

If the system restarts automatically, it bootstraps itself into memory from the system disk. After the system is bootstrapped into memory, control passes to the initialization routines. The system recognizes startup was through an automatic restart rather than through normal startup. Consequently, the system initializes itself in automatic restart mode.

If an error trap occurs within one minute of system startup or if two error traps occur within the same minute, the system halts at address 54. During automatic restart the system protects itself against an infinite loop of error traps and automatic restarts caused by a repeating hardware malfunction.

2.3.3 Automatic Restart Mode Initialization

When the system is initialized in automatic restart mode, control bypasses all parts of the initialization code that call for operator intervention and initializes the system using parameters specified at previous startup. The system:

1. Logs job 1 in to the system on KB0: under account [1,2]
2. Invokes [0,1]SYSINI.COM
3. Executes [0,1]START.COM with the P1 argument set to 'CRASH' rather than 'START'
4. Performs specific operations appropriate for an automatic restart

2.3.4 Restarting with a Different Monitor

You can, if you like, keep several different monitors on your system. When you start up timesharing, the system prompts you for your choice of monitor:

```
Monitor name to start <default name>?
```

You can also pre-set the choice of monitor so that the new monitor starts working at the next system shutdown and reboot. Use the SET SYSTEM/MONITOR_NAME command to do this. For example:

```
$ SET SYSTEM/MONITOR_NAME=ALTRNT
```

This command causes the system to use the monitor stored in the file ALTRNT.SIL at the next automatic restart. ALTRNT.SIL also becomes the default name for the next time you start up timesharing manually. You need the SYSIO privilege to use the SET SYSTEM/MONITOR_NAME command.

If you have the WREAD privilege and use the SHOW SYSTEM command, it displays the name of the monitor if it is different from the current monitor name.

2.3.5 Restarting with a Different Swap Size

If your system has I&D space, you set the maximum size of jobs when you install the system. You can change the maximum size with the SET SYSTEM/SWAP_MAXIMUM command. For example:

```
$ SET SYSTEM/SWAP_MAXIMUM=40
```

This command sets the maximum job size at 40K words. You can reset the maximum size to any value between 32K and 64K. The change in size does not take effect until the next time you start the system. You need the TUNE privilege to use the SET SYSTEM/SWAP_MAXIMUM command.

If you have the TUNE privilege and use the SHOW SYSTEM command, it displays the new value of the swap maximum if it is different from the current swap maximum.

2.3.6 Power-Fail Hardware Used by RSTS/E

When power fails, a PDP-11 processor traps through the vector at location 24. When power is restored, you can configure the system to do one of three things:

- Halt.
- Boot a specific device.

  If the hardware is configured to boot the system disk, INIT.SYS prints its identification line and the Start timesharing? <Yes> prompt and waits for yoı to enter one of the options from Table 2–1.
- Trap again through the vector at location 24.

  In this case, the system must have core memory or battery backup. If memor contents are still valid (which means the battery backup is not exhausted), RSTS/E performs an automatic restart.

Chapter 3

The System Startup Command Files

This chapter describes how you control system startup with the system startup command file (START.COM). Startup occurs when you answer YES to the Start timesharing? <Yes> prompt, use the START option of the initialization code, or when the system restarts automatically.

At startup, the monitor executes the command file [0,1]SYSINI.COM which in turn executes [0,1]START.COM. You can modify START.COM (but not SYSINI.COM) to suit the requirements of your installation.

To control startup efficiently, you should understand the conditions at startup time:

- Login attempts are prohibited (the monitor disables the login capability)
- The monitor logically mounts only the system disk read/write
- No output is made to any terminal (except the console)
- The monitor logs the console terminal KB0: into the system under the system library account [1,2]
- The monitor assigns ALL privileges to this initial job regardless of the privileges assigned to account [1,2]
- The monitor loads the DCL run-time system
- The monitor starts execution of the command file _SY0:[0,1]SYSINI.COM at the console

3.1 Creating and Using Command Files

The SYSINI.COM command file contains commands necessary to initialize your RSTS/E system properly. Do not edit this file in any way; this command file is reserved by DIGITAL and is replaced with every update of the system.

The START.COM file that comes with the RSTS/E distribution kit is a sample only; it may not execute properly on your system without modification. Make sure START.COM includes the necessary commands to initialize your installation. Use a text editor on your system to modify the sample START.COM file supplied with the RSTS/E kit.

NOTE

Be sure both command files are in account [0,1] on the system disk.

The startup control file should perform the following operations:

- Mount public and private disks
- Assign system-wide logicals
- Run the ERRINT program to start error logging
- Enable data caching (if selected at system installation)
- Set default system characteristics
- Install run-time systems
- Install libraries
- Set terminal characteristics
- Define system commands
- Set printer characteristics
- Start Print/Batch Services (PBS) package
- Start OPSER-based spooling package (optional)
- Enable Local Area Transport (LAT) (optional)
- Start error analysis if needed
- Enable logins
- Logout the initial startup job

The order in which the operations are performed is critical. For example, mounting all public disks ensures the integrity of the public structure by making available all files on the system. You can mount any private disks in a similar manner. The various steps tailor the system according to your installation's requirements.

3.2 System Startup Command File: START.COM

A complete system startup command file contains the required commands to perform initialization for timesharing and to perform routine operations. The following sample shows a startup command file:

```
$ !
$ ! START.COM
$ !
$ !   Copyright (c) 1988 by
$ !  Digital Equipment Corporation, Maynard, MA.
$ !
$ ! P1 - Type of startup (either START or CRASH)

$ ! *** Prompt to continue ***
$ !
$ _if P1 .eqs. "" then P1 = "START"  ! Default is START
$ S = "system startup"
$ _if P1 .eqs. "CRASH" then S = "crash restart"
$ GO = ""
$ _inquire/nopunc/exit=END/time_out=10 GO "Proceed with ''S'? <YES> "
$ _if f$instr(1,"YES",f$edit(GO,-2)) .eq. 1 then _goto START

$END:
$ _exit
```

```
$START:
$ _open/logfile/replace  'P1'.LOG
$ _set noecho/warning
$ _set verify
$ _write 0 ""
$ _write 0 " Beginning RSTS/E ''S'..."
$ _set noon

$ ! *** Install Monitor Overlays ***
$ !
$ _write 0 f$time()+ "   Installing monitor overlays"
$ _run $UTLMGR.TSK
!load/overlay ATTRIBUTE
!load/overlay DCL
!load/overlay DELETE_RENAME
!load/overlay DIRECTORY
 load/overlay SYSTEM_CALLS
!load/overlay TERMINAL
$ _eod

$ ! *** Install Virtual Disk ***
$ !
$! _write 0 f$time()+ "   Installing Virtual Disk"
$! _create/virtual 200
$ ! Create a 200 block virtual disk
$ ! This will use 50KW of memory

$ ! *** Mount Disks ***
$ !
$ S = ""
$ _if P1 .eqs. "CRASH" then S = "and rebuilding "
$ _write 0 f$time()+"   Mounting ''S'disks"
$ !  (If this is a crash recovery, MOUNT will rebuild
$ !     the disks if necessary.)
$ !   examples:
$ !   $ mount/nowrite DR3:LIBRAR L:
$ !   $ mount  DB2:LOOKUP

$ ! *** Assign (or Replace) Systemwide Logicals ***
$ !
$ _write 0 f$time()+"   Assigning logical names"
$ !   examples:
$ !   $ assign/system         _SY:[2,9]  FERN:
$ !   $ assign/system/replace _DB2:[0,3] ERROR$:

$ ! *** Start Error Logging ***
$ !
$ _write 0 f$time()+ "   Starting error logging"
$ _set job/priority=0
$ _if P1 .nes. "CRASH" then _goto CONT1
$ _run ERROR$:ERRINT
100
YES
$ _EOD
$ _goto CONT2

$CONT1:
$ _run ERROR$:ERRINT
100
NO
$ _EOD

$CONT2:
$ _set job/priority=-8
$ ! *** Install Auxiliary Swap Files ***
$ !
$ ! If you have additional auxiliary swap files (SWAP0 or SWAP3) to
$ ! install, add them here.
$ !
$ ! Note: SWAP1.SYS already installed if present.
$ !
$ !   examples:
$ !   $ _write 0 f$time()+ "   Installing auxiliary swap files"
$ !   $ _install/swapfile=0 _SY0:[0,1]SWAP0.SYS
$ !   $ _install/swapfile=3 _SY0:[0,1]SWAP3.SYS
$ ! *** Enable Caching and Set Default System Characteristics ***
$ !
$ _write 0 f$time()+ "   Setting system characteristics"
$ _run $UTLMGR.TSK
 set cache/file/enable/keep=15/clustersize=2
 set system/date_format=alphabetic
 set system/time_format=am_pm
 set system/label=dos
 set system/density=1600
 set system/name="RSTS/E V9.7"
 set system/powerfail_delay=300
 set system/hangup=delayed
 set system/password_prompt=(network,dialup)
$ _eod
$ ! *** Install Run-time Systems and Libraries ***
$ !
$ ! Note: Resident library CSPLIB already installed.
$ !
$ _write 0 f$time()+ "   Installing run-time systems and libraries"
$ _run $UTLMGR.TSK
 install/runtime_system BASIC
 install/runtime_system TECO$:TECO
 install/library/noaddress=unrestricted EDT$:EDT
 install/library/address RMS$:RMSRES
 install/library/noaddress RMS$:RMSLBA
 install/library/noaddress RMS$:RMSLBB
 install/library/noaddress RMS$:RMSLBC
 install/library/noaddress RMS$:RMSLBD
 install/library/noaddress RMS$:RMSLBE
 install/library/noaddress RMS$:RMSLBF
 install/library/address RMS$:DAPRES
$ _EOD
$ ! *** Set Terminal Characteristics ***
$ !
$ _write 0 f$time()+ "   Setting terminal characteristics"
$ _load/overlay TERMINAL
$ _run $TERMGR.TSK
 set terminal kb0:/permanent/device_type=LA34
 ! System console as hard copy
 set terminal kbd0:/permanent/device_type=LA100
 ! Pseudo-keyboard 0 for batch
!set terminal kbf0:/permanent/noautobaud/speed=9600/device_type=VT100
 ! DH0 line 0 as VT100 at 9600 baud
!set terminal kbg0:/permanent/noautobaud/speed=9600/device_type=VT100
 ! DZ0 line 0 as VT100 at 9600 baud
!set terminal kbg8:/permanent/noautobaud/speed=9600/device_type=VT240
 ! DZ1 line 0 as VT240 at 9600 baud
$ _EOD
$ _unload/overlay TERMINAL
$ ! *** Define System Commands ***
$ !
$ _write 0 f$time()+ "   Defining system commands"
$ ! Install system commands (CCLs)
$ _run $UTLMGR.TSK
 define/command/system BCK-       RMS$:RMSBCK.TSK
!define/command/system BPC-REF    AUXLIB$:BPCREF.TSK /line_number=CCL
 define/command/system BYE-       $LOGOUT.TSK        /privilege
 define/command/system CNV-       RMS$:RMSCNV.TSK
 define/command/system DI-RECTORY $DIRECT.TSK        /line_number=CCL/privilege
 define/command/system DEF-       RMS$:RMSDEF.TSK
 define/command/system DES-       RMS$:RMSDES.TSK
 define/command/system DSP-       RMS$:RMSDSP.TSK
 define/command/system EDT-       EDT$:EDT.TSK
!define/command/system FIT-       AUXLIB$:FIT.TSK    /line_number=CCL/privilege
 define/command/system IFL-       RMS$:RMSIFL.TSK
 define/command/system LBR-       $LBR.TSK
 define/command/system LIBR-      $LIBR.SAV          /line_number=8208
 define/command/system LIN-K      $LINK.SAV          /line_number=8208
 define/command/system MACR-O     $MACRO.SAV         /line_number=8216
 define/command/system MAC-       $MAC.TSK
!define/command/system MAK-E      TECO$:TECO.TEC     /line_number=3584
!define/command/system MU-NG      TECO$:TECO.TEC     /line_number=3584
 define/command/system PAT-       $PAT.TSK
 define/command/system PIP-       $PIP.SAV           /line_number=8208/privilege
 define/command/system RST-       RMS$:RMSRST.TSK    /privilege
 define/command/system SW-ITCH    $SWITCH.TSK        /line_number=CCL/privilege
 define/command/system SY-STAT    $SYSTAT.TSK        /line_number=CCL/privilege
!define/command/system TE-CO      TECO$:TECO.TEC     /line_number=3584
 define/command/system TKB-       $TKB.TSK
$ _EOD

$ ! *** Set Printer Characteristics ***
$ !
$ _write 0 f$time()+ "   Setting printer characteristics"
$ !   examples:
$ !   $ _set printer LP0:/device_type=LP11
$ !   $ _set printer LP1:/device_type=LA180
$ !   $ _set printer LP2:/device_type=LN01

$ ! *** Start Spoolers ***
$ !
$ _write 0 f$time()+ "   Starting spoolers"
$ ! Start PBS spooling package
$ _start/queue/manager
$ !
$ !  ** OPTIONAL** Start OPSER (old) spooling package
$ !
$ _goto CONT3  ! If OPSER is desired, delete this line.
$ ! Install related system commands
$ _define/command/system PL-EASE   OPSER$:PLEASE.TSK  /privilege/line_number=CCL
$ _define/command/system QU-EUE    OPSER$:QUE.TSK     /privilege/line_number=CCL
$ _run OPSER$:OPSER
LOG OPSER$:OPSER.LOG;ALL
CHA KB0:
MESS ALL
OPER KB*:[1,2]
DETACH
$ _run OPSER$:QUEMAN
DETACH/PRI:0/RUN:6
$ _set job/priority=-8
$ _run OPSER$:SPOOL
LP0:/HEAD:2/RUN:9/WIDTH:80
$ _set job/priority=-8
$ _run OPSER$:SPOOL
LP1:/HEAD:1/RUN:9/WIDTH:132
$ _set job/priority=-8
$ _run OPSER$:BATCH
BA0:/ERROR:FATAL/NOQUEUE/NODELETE
```

```
$ _set job/priority=-8
$ _run OPSER$:BATCH
BA1:/ERROR:FATAL/NOQUEUE/NODELETE
$ _EOD

$CONT3:

$ ! *** Start ANALYS if in Crash Recovery ***
$ !
$ _if P1 .nes. "CRASH" then _goto CONT4
$ _write 0 f$time()+ "   Starting analysis"
$ _run ERROR$:ANALYS
[0,1]CRASH.SYS
ERROR$:ANALYS.DMP
ERROR$:ERRCRS.FIL/DET
$ _EOD

$CONT4:

$ ! *** Starting DECnet/E V4.0 ***
$ !
$! _write 0 f$time()+ "   Starting DECnet/E V4.0"
$ !  First define the DECnet related CCL commands
$! _define/command/system NCP DECNET$:NCP.TSK/priv/line=ccl
$! _define/command/system NFT DECNET$:NFT.TSK/priv/line=ccl
$! _define/command/system NET DECNET$:NET.TSK/priv/line=ccl
$! _define/command/system TLK DECNET$:TLK.TSK/priv/line=ccl
$ !  Now start the network
$! ncp set system
$! ncp set exec state on
$! ncp set cir una-0 state on

$ ! *** Start up LAT ***
$ !
$! _write 0 f$time()+ "   Starting LAT"
$! _SET NODE/LAT/ENABLE=0
$! _SET NODE/LAT/IDENTIFICATION="RSTS/E V9.7 LAT"
$! _CREATE/SERVICE/LAT RSTS/IDENTIFICATION="Unauthorized use prohibited"
$! _START/LAT XE0:

$ ! *** Enable Logins ***
$ !
$ _set system/logins
$ _broadcast/all "RSTS/E is on the air..."
$ _close/logfile
$ _set noverify
$ _set echo
$ _logout/br
$ _exit
```

NOTE

The startup control files must be in [0,1] on the system disk (_SY0:, the disk that was bootstrapped).

3.2.1 Mount All Disks

To ensure the integrity of the public structure by making available all files on the system, you should mount all public disks. You can mount any private disks in a similar manner. If this is a restart from a system crash, MOUNT rebuilds the disks as required. See Chapter 15 for more information on mounting and initialization of disks.

3.2.2 Assign Systemwide Logical Names

You can assign a systemwide logical name to a specified device type, unit numbe and PPN (optionally) with the ASSIGN/SYSTEM command. See Chapter 12 for more information on the use of systemwide logical names.

3.2.3 Start Error Logging

It is important that you run the ERRINT program on your system. This progran sets initial conditions for system error logging and runs the ERRCPY program which transfers to a disk file all errors logged by the RSTS/E monitor. If ERRIN is the first program to be run (as DIGITAL recommends), it will occupy job number 1 on the system and thus be easily monitored. Chapter 10 describes ERRINT and ERRCPY.

NOTE

> If your system has more than 256K words of memory, you should consider adding /LOCK to the first line after the RUN ERRINT command lines to keep ERRINT resident at all times.

3.2.4 Enable Caching (Optional)

The SET CACHE/FILE/ENABLE command enables caching of data files whose UFD entry or OPEN MODE specifies caching. The /KEEP qualifier lets you spec ify the time that an unused cache cluster is kept in memory. The /CLUSTERSIZ qualifier controls the number of contiguous data blocks that are copied from the disk to the cache whenever file data is cached. For more information on data an directory caching, see the section "SET CACHE Command" in Chapter 11.

3.2.5 Set Default System Characteristics

The SET SYSTEM command sets the default system characteristics. In the sample START.COM, for example:

- The date format is alphabetic
- The time format is AM_PM
- The magnetic tape label is DOS
- The magnetic tape density is 1600 bpi
- The system name is EDERE
- The power fail delay is 300 seconds
- NETWORK and DIALUP users must enter the system password

See Chapter 14 for more information on setting system default characteristics.

3.2.6 Install Run-Time Systems

The INSTALL/RUNTIME_SYSTEM command installs auxiliary run-time systems (like BASIC) in memory. See the section "Run-Time System Control" in Chapter 12 for more information.

3.2.7 Install Libraries

The INSTALL/LIBRARY command installs resident libraries (like RMSRES) in memory. See Chapter 12 for more information on resident library control.

3.2.8 Set Terminal Characteristics

At startup time, the system sets the characteristics of all keyboard lines (except line number 0) to hard copy, 80 column output at the line speed of 300 baud. Thus, you need to use the SET TERMINAL command to set the correct characteristics of your terminals. Chapter 8 contains detailed descriptions of the SET TERMINAL command qualifiers.

3.2.9 Define System Commands

It is most convenient to define Concise Command Language (CCL) system commands on the system at the start of timesharing using the DEFINE/COMMAND/SYSTEM command. The abbreviations shown in the definitions of the CCL commands reflect the way the RSTS/E documentation describes the commands. As the system manager you can redefine the abbreviations. You should not alter the fully defined command because each program is coded to recognize the fully expanded command.

Note that the abbreviation point should be the same as the one used by DCL itself. For most DCL commands, the abbreviation point follows the second character, but for some, the abbreviation point follows the third or fourth character, to assure a unique abbreviation for each command.

Use caution when you define system commands in this way. You may make features of other keyboard monitors unavailable. This may happen if, for example, you define PRINT as a CCL command. BASIC-PLUS users will then not be able to use the immediate mode PRINT statement.

Two CCL commands duplicate commands the BASIC-PLUS keyboard monitor recognizes: HELLO and BYE. The commands are shown here to point out what BASIC-PLUS does by default. Other keyboard monitors may not recognize the commands HELLO and BYE if they do not exist as CCL commands.

Each CCL definition requires one small buffer. The buffer will be allocated from the FIP small buffer pool. If this buffer pool is full, however, the monitor allocates the buffer for the CCL definition from the general buffer pool. See Chapter 12 for more information on the DEFINE/COMMAND/SYSTEM command.

3.2.10 Set Printer Characteristics

Just as you did for terminals, you need to set the characteristics of your printers. Minimally, you need to specify the device type of each printer on your system. See Chapter 11 for more information about the qualifiers for the SET PRINTER command.

3.2.11 Start Print/Batch Services

The Print/Batch Services package (PBS) provides printing services for line printers or terminals, and batch processing. The package is started with a singl command. Chapter 9 contains detailed descriptions of the DCL commands you can use to manage print/batch services.

3.2.12 Start OPSER-Based Spooling Package (Optional)

The OPSER-based spooling package, like PBS, provides print and batch services RSTS/E offers this optional package for compatibility with previous versions of RSTS/E. New applications should use PBS. For more information on OPSER, see Appendix A of this manual and the *RSTS/E Utilities Reference Manual.*

3.2.13 Enable LAT (Optional)

A new sample startup file is installed in Versions 9.6 and later. If you already have a [0,1]START.COM file, then the sample uses the version number as its extension, for example [0,1]START.096. If you do not have a [0,1]START.COM file, then the sample is installed on your system with that name. The sample ha commands in it to enable LAT, but these are in the form of comments. If you wan to enable LAT automatically on system startup, edit the [0,1]START.COM file an remove the exclamation points (!) from the LAT commands. The commands are:

```
$ ! *** Start up LAT ***
$ !
$! _write 0 f$time()+ "   Starting LAT"
$! _SET NODE/LAT/ENABLE=0
$! _SET NODE/LAT/IDENTIFICATION="RSTS/E V9.7 LAT"
$! _CREATE/SERVICE/LAT RSTS/IDENTIFICATION="Unauthorized use prohibite
$! _START/LAT XE0:
```

3.2.14 Start Error Analysis

Run the ANALYS program immediately after a system crash to recover valuable diagnostic information. See Chapter 10 for a description of the ANALYS program

3.2.15 Enable Logins

The SET SYSTEM/LOGINS command should appear after the commands that perform all required and routine operations. The SET SYSTEM/LOGINS command lets users log in to the system. You can then notify users that timesharing has started. See Chapter 13 for more information on job management.

3.2.16 Logout the Initial Startup Job

You should always logout the initial startup job, since it runs with ALL privileges. Failing to do so can create a security risk if the system console is left unattended.

Chapter 4

Shutting Down the RSTS/E System

4.1 Performing System Shutdown: SHUTUP

The shutdown procedures for the RSTS/E system are critically important. If you are not careful in performing system shutdown, valuable user data can be irretrievably lost. To understand shutdown procedures fully, you need to understand other RSTS/E system procedures. Chapter 12 describes these concepts such as controlling logins, removing system files and resident libraries, sending messages to all logged-in users, and forcing commands to unresponsive terminals.

4.1.1 Running SHUTUP

You use the SHUTUP program to shut the system down in an orderly manner. SHUTUP is in the system library account $ ([1,2]). SHUTUP must be installed as a privileged program (protection code <232>). If not, the system displays the following error message:

```
?Program must be privileged
```

To run SHUTUP, you must have SHUTUP pnvilege. If you do not, the system displays the following error message:

```
?SHUTUP privilege required
```

SHUTUP can be run from any local terminal over a dialup line or over a LAT line. SHUTUP cannot be run from a pseudo keyboard (for example, by a SET HOST connection).

To verify that conditions are suitable for shutting down the RSTS/E system, use any of the SHOW commands, such as:

- SHOW JOBS
- SHOW DISKS
- SHOW DEVICES/ALLOCATED

By executing these SHOW commands, you can determine what jobs are active on the system and what disks and other peripheral devices are in use.

4.2 SHUTUP Phases

As SHUTUP runs, it proceeds through the following 14 phases:

1. Set-up dialogue
2. Warning message
3. DECnet/E shutdown (if necessary)
4. Initial job killing
5. Print/Batch Services (PBS) shutdown (if necessary)
6. OPSER (optional spooling package) shutdown (if necessary)
7. EVTLOG shutdown (if necessary)
8. ERRCPY shutdown (if necessary)
9. Final job killing (if necessary)
10. EMT logging shutdown (if necessary)
11. Unload and remove run-time systems and resident libraries
12. Log file closing
13. Swap file removal
14. Disk dismount
15. Final shutdown

The following sections describe these phases, the operator interactions required, and the expected results of the shutdown operation. Possible results, which you may not expect are also described. Several sample runs of SHUTUP show the results of shutting down the system.

4.2.1 Set-Up Dialogue Phase

The set-up dialogue phase processes all questions relating to the selection of options in running SHUTUP. It also checks for the presence of PBS or OPSER.

In response to questions that appear during this phase, you can press the circumflex (^) key to return to the previous question. Otherwise, you should terminate your responses by pressing the Return key.

If the Print/Batch Services (PBS) package is present, SHUTUP prints the following message:

```
Allow Print/Batch entries to complete <YES>?
```

If you type NO, SHUTUP aborts all PBS entries that are in progress. If you press Return or type YES, SHUTUP lets all currently running PBS entries finish.

If OPSER is present and running, SHUTUP prints the following question:

```
Use OPSER for utilities shutdown <YES>?
```

SHUTUP operates in either of two modes:

- Without OPSER: In this mode, regardless of whether OPSER is running, the SHUTUP program treats all jobs in the system alike during the shutdown procedure (except itself, the Print/Batch Services program, the DECnet/E EVTLOG program, the EMT logger program, and ERRCPY).
- With OPSER: In this mode SHUTUP allows OPSER to shut down its controlled jobs before SHUTUP continues with the regular shutdown procedures.

You should be familiar with the OPSER program and how it controls various utility programs (such as QUMRUN and the OPSER spooling package programs).

Press Return or type YES to have SHUTUP communicate with OPSER during the shutdown procedures. If you type NO, SHUTUP prints another question to confirm your response:

```
Are you sure you don't want to use OPSER <NO>?
```

Pressing Return or typing NO causes the program to repeat the previous question so you can again decide to use or not to use OPSER during shut down. Otherwise, the program ignores OPSER during the shutdown procedures.

If you choose to use the OPSER program, SHUTUP prints the following additional question:

```
Allow utilities to reach logical end point <YES>?
```

If you press Return or type YES, the program directs OPSER to tell its online jobs to shut down their operations at the next logical breakpoint in their job streams. For a line printer spooler, for example, this breakpoint occurs between queued jobs.

If you type NO, the program directs OPSER to tell its online jobs to abort and stop operations immediately. You should abort in this way only in an emergency: jobs being processed at the time of shutdown may be lost unless you have previously halted queuing and spooling operations and requeued these jobs.

SHUTUP then asks you to specify the number of minutes you want to elapse before shutdown occurs:

```
Minutes until system shutdown (0-99) <5>?
```

This waiting period is called the "warning message phase" and may be as short as 0 minutes or as long as 99 minutes. Press Return to select the default value of 5 minutes. If you decide to bring the system down immediately by typing 0, SHUTUP does not ask the two questions about disabling logins and the DECnet/E network.

When you specify a shutdown period greater than 0 minutes, SHUTUP allows you to select the amount of time that will elapse before further logins are disabled:

```
Minutes until logins are disabled (0-99) <0>?
```

This question gives you the opportunity to select the amount of time during which users can continue to log in to the system as shutdown occurs. Users can log in to the system (as if the system were not shutting down) for the number of minutes you specify. The ability to allow users this flexibility is useful when you specify a long shutdown period.

There will be times when you do not want to allow any more users on the system. In this case, accept the default response by pressing Return. Then, users can no longer log in to the system; however, those already logged in remain unaffected.

NOTE

> SHUTUP disables all logins at least one minute before the initial job killing phase.

The last message in the set-up dialogue phase allows you to choose the number of minutes that will elapse before new network activity is disabled:

```
Minutes until new network activity is disabled (0-99) <nn>?
```

The number of minutes SHUTUP selects for the default value <nn> depends on your response to the system shutdown question. As in the previous question, being able to specify when the network will shut down allows users time to complete network tasks. If you choose to allow no further logical links to be created, type 0, and then press Return. Those still using the network can continue to work but no new users are allowed access to the network.

```
Shutdown with automatic RESTART <NO>?
```

If you type YES, you will automatically restart the system after the shutdown. The system does not restart from the terminal you are using. INIT.SYS uses the console terminal (KB0:), no matter what terminal you use for SHUTUP. You must determine whether INIT.SYS failed to start by looking at the console terminal.

Use automatic restart to change the monitor name, the swap maximum, LAT support, or the maximum number of pseudo keyboards.

4.2.2 Warning Message Phase

At the beginning of the warning message phase, as well as in the beginning of all other phases that SHUTUP may execute, a message appears on the system console terminal (KB0:) in the following format:

```
hh:mm AM dd-mmm-yy  ########  <phase title>  ########
      PM
```

SHUTUP prints:

- The time that it issues the message in the format hh:mm {AM | PM}
- The date that it issues the message in the format dd-mmm-yy
- The title of the current phase

SHUTUP prints the following message on the system console terminal when the time specified for disabling logins has elapsed:

```
Further LOGINs are now disabled
```

The RSTS/E shutdown procedure requires that logins are disabled before the monitor can perform the final shutdown operation. During the following phases, SHUTUP continuously checks that logins remain disabled. If, for any reason, logins are not disabled, SHUTUP immediately aborts all operations with the following message:

```
?Logins not disabled
??SHUTUP aborting - please try again later
```

You should determine the cause of the change, correct the problem, and run SHUTUP again.

If your system has DECnet/E support, the SHUTUP program prints the following message when the time specified for disabling network activity expires:

```
Further network activity is now disabled
```

The message tells you that no new logical links can be formed. Those links that are already active, however, can remain active until the time to network shutdown has elapsed. Before the network shuts down, the DECnet/E program EVTLOG may print messages indicating the status of the network. The initial EVTLOG message marks the beginning of the network shutdown phase. The second message, issued later in the shutdown process, indicates that the network has been shut off completely. See the section "SHUTUP Example: DECnet/E Shutdown" for an example of these messages.

During the warning message phase, the program issues warning messages indicating the system is shutting down. These messages appear on most keyboards in the system, including pseudo keyboards. The following keyboards are the exceptions:

OSC - The OPSER operator services console
KBn: - The terminal running shutup

The program issues warnings after an appropriate waiting period, defined by the following formula:

waiting time = (total time left)/5 + 1 minutes

For an initial waiting period of 60 minutes, warning messages appear at 60, 47, 37, 29, 23, 18, 14, 11, 8, 6, 4, 3, 2, and 1 minute before shutdown begins. SHUTUP prints the following message on all system terminals at each warning time:

```
hh:mm AM dd-mmm-yy  System going down in n minutes, please finish up
      PM
```

At the same time, the following message appears on the terminal running SHUTUP:

```
n minute message sent
```

When the time to shutdown has expired or if the initial waiting period is specified as 0, the final warning message appears:

```
hh:mm AM dd-mmm-yy **** FINAL WARNING!!!!! System shutting down ****
      PM
```

At this point, the program enters the next phase. SHUTUP prints the time and date for both the SYSTEM GOING DOWN and the FINAL WARNING messages.

4.2.3 DECnet/E Shutdown Phase (When Necessary)

The time to DECnet/E shutdown begins as soon as you enter the number of minutes that should elapse before new network activity is disabled:

```
Minutes until new network activity is disabled (0-99) <nn>?
```

The number of minutes, shown in angle brackets <nn>, is the default response. It assumes the same value you typed in response to the SHUTUP question:

```
Minutes until system shutdown (0-99) <nn>?
```

Rather than accept the default response (that is, to the network activity question) and have new network activity disabled at the same time the system shuts down, it is often better to disable new network activity before shutdown occurs. Entering a value less than the number of minutes to shutdown establishes a period before shutdown in which no new links can be created; it also allows time for those already active to complete network tasks. If you want new network activity disabled immediately, type 0 and press Return. When the time to disable network activity has elapsed (0 to 99 minutes), SHUTUP prints a message to indicate no further network activity is allowed:

```
Further network activity disabled
```

SHUTUP disables new network activity by executing the equivalent of the Network Control Program (NCP) command SET EXECUTOR STATE SHUT. This prevents the creation of new logical links but does not affect existing logical links. In reality, a user can create new network links until the network activity is disabled. After that period ends, no new links can be created, but users already on the network can continue working until the system shuts down. The network remains up until there are no links active or until the system shuts down.

As soon as all logical links have been disconnected by their users, the network state changes automatically to OFF. The state is immediately set to OFF if no links existed at the time SHUTUP disabled new network activity. If the DECnet/E event logging program EVTLOG is on and has event 2.0 enabled, it logs both the transition to SHUT and the one to OFF. (See the *DECnet/E System Manager's Guide* for a description of EVTLOG.)

SHUTUP never sends special messages indicating the approach of a DECnet/E shutdown. Users can tell that SHUTUP is shutting down the system only from SHUTUP messages to all terminals and from the fact that the network is in the SHUT instead of the ON state.

If the network has still not completely shut down at the end of the warning message phase (because active links remain), SHUTUP explicitly sets the network to OFF. This action disconnects all remaining logical links. SHUTUP then waits 15 seconds to allow affected jobs to complete network tasks and exit, before it enters the job killing phase.

If you decide not to shut the system down (thus, not shut off the network) after you have started SHUTUP, entering a Ctrl/C to stop the execution of SHUTUP may reactivate the network. A Ctrl/C causes SHUTUP to print a message telling you the state in which the network was left. If the network is not in the state you want, you must use NCP to place the network and the event logger (EVTLOG) in the desired state. See the *DECnet/E System Manager's Guide* for a description of NCP.

4.2.4 Initial Job Killing Phase

In the initial job killing phase, SHUTUP begins to clear the system of currently active jobs. If SHUTUP finds there is only one job (itself) running on the system, it skips this phase and then continues the shutdown procedure at the unload and remove run-time systems and resident libraries phase.

If SHUTUP is using PBS in the next phase, during this phase the program ignores all jobs currently associated with PBS (as indicated by entries in PBS's on-line job table). The program also ignores any job whose primary keyboard is a pseudo keyboard. Such a job is probably being controlled by one of the batch servers.

SHUTUP further divides all other jobs active in the system into two classes: attached and detached. This classification depends on whether they have an attached primary keyboard (KB:). For all attached jobs, SHUTUP forces the following text string to the keyboard input buffer:

```
^C  $_LOGOUT/BRIEF
```

For all detached jobs, SHUTUP issues the Kill job SYS call (SYS 8) to remove the job.

SHUTUP makes two passes through the current active job table during this phase. In the first pass, the program terminates all active jobs not being ignored At the end of the first pass, it establishes the number of attached jobs that were found and forced to logout. SHUTUP then waits for a specified period to allow the LOGOUT program to complete operations on each of the attached jobs.

After the waiting period (if any) expires, the program makes a second pass through the active job table. During this pass, all jobs not being ignored are removed with the kill job SYS call. At the end of the pass, the program checks the table. If any jobs that should have been killed were not, SHUTUP aborts operations with the following error message:

```
?SHUTUP failed in initial job killing phase
??SHUTUP aborting - please try again later
```

You should determine why SHUTUP could not kill all jobs, correct the problem, and run SHUTUP again.

If SHUTUP successfully removes all jobs during the phase, or if at any time during either pass the number of active jobs becomes one (SHUTUP only), the next phase begins.

4.2.5 Print/Batch Shutdown Phase (When Necessary)

The program skips the Print/Batch Services (PBS) shutdown phase if PBS is not running on the system. If PBS is not running, SHUTUP continues with the next phase; no message is displayed.

If PBS is running, SHUTUP sends a message to QMAN telling QMAN to shutdown PBS. QMAN acknowledges the shutdown command by sending back a confirmation message. If SHUTUP does not receive the confirmation message within two minutes after the offline request was sent, SHUTUP aborts the shutdown operation and displays the following:

```
?No response from Print/Batch Services after 2 minutes
```

Otherwise, SHUTUP displays the number of entries currently processing in PBS in the following format:

```
Print/Batch Services shutting down -- # jobs are completing.
Print/Batch Services shutting down -- # jobs are being terminated.
```

Then SHUTUP waits for PBSMGR to shutdown PBS. The maximum wait period is one hour if jobs are allowed to complete and one minute if jobs are aborted. If shutdown occurs in the specified time, SHUTUP displays the following message and goes to the next phase:

```
Print/Batch Services shutdown complete at hh:mm am/pm
```

Otherwise, SHUTUP aborts and displays the following error message:

```
?Print/Batch Services shutdown taking too long
??SHUTUP aborting - please try again later
```

4.2.6 OPSER Shutdown Phase (When Necessary)

The program skips the OPSER shutdown phase if:

- OPSER is not running on the system
- You elect not to use OPSER during the shutdown procedures

If you choose to use OPSER, SHUTUP immediately detaches from the terminal running SHUTUP. OPSER reattaches to KB0: for use during OPSER's shutdown procedures. As the detach occurs, the following messages appear:

```
DETACHING...
'OPSER' ATTACHING
#
```

SHUTUP sends a message to OPSER directing it to begin its shutdown procedures by selecting the appropriate mode of OPSER shutdown. SHUTUP then waits 60 seconds to see if OPSER receives the message and/or is functioning properly. If OPSER does not respond properly within that time, SHUTUP attempts to reattach to the terminal running SHUTUP. If the operation is successful, the following message appears:

```
?OPSER not active
??SHUTUP aborting - please try again later
```

If the terminal is attached or assigned, SHUTUP waits for one second and tries again. You should take whatever steps are necessary to free the terminal running SHUTUP for use so SHUTUP can complete its activities.

If OPSER is shutting down its online programs in the immediate mode, it sends each program the appropriate message to stop operations. OPSER pauses to make sure all programs have completed their assignments. OPSER then closes its files, prints the following message on the system console (KB0:), and stops execution:

```
'OPSER' TERMINATING
```

SHUTUP allows OPSER 120 seconds to complete its shutdown in the immediate shutdown mode. If unsuccessful, SHUTUP aborts its operations, signaling the failure as follows:

```
?OPSER shutdown taking too long
??SHUTUP aborting - please try again later
```

If OPSER is shutting down in the logical end mode, it proceeds to shutdown successive levels in its online job table. Appropriate system console messages signal the end of each job (see "SHUTUP Operation Examples"). When all online OPSER jobs are gone, OPSER closes its files and kills itself, freeing the system console for SHUTUP use again. OPSER tells you when it finishes processing with the message:

```
'OPSER' TERMINATING
```

SHUTUP allows 60 minutes in the logical end mode for OPSER to complete operations. If the procedure is not successful, SHUTUP aborts operations with the error message described previously. Otherwise, the shutdown of OPSER-related jobs and activities is complete and the program enters the next phase.

4.2.7 EVTLOG Shutdown Phase (When Necessary)

The SHUTUP program enters the EVTLOG shutdown phase if it finds EVTLOG in the message/receiver table. (DECnet/E does not need to be active for SHUTUP to perform this phase.) SHUTUP sends EVTLOG a special message that causes EVTLOG to log the shutdown occurrence on the system console, perform various other shutdown operations, and kill itself.

When SHUTUP detects that EVTLOG has completed its shutdown activities and is no longer present, SHUTUP proceeds to the next phase. SHUTUP allows 60 seconds for EVTLOG to complete its tasks, and if failure occurs, SHUTUP aborts operations with the following error message:

```
?EVTLOG failed to shutdown
??SHUTUP aborting - please try again later
```

See the *DECnet/E System Manager's Guide* for information about the EVTLOG program.

4.2.8 ERRCPY Shutdown Phase (When Necessary)

The program enters the ERRCPY shutdown phase if the system error logging utility ERRCPY is present. SHUTUP sends ERRCPY a special message which causes ERRCPY to log the shutdown occurrence in the system error log, close the file, and kill itself.

When SHUTUP detects that ERRCPY has completed its shutdown and is no longer present, SHUTUP proceeds to the next phase. SHUTUP allows 60 seconds for ERRCPY to complete its tasks. If failure occurs, SHUTUP aborts operations with the following error message:

```
?ERRCPY failed to shutdown
??SHUTUP aborting - please try again later
```

4.2.9 Final Job Killing Phase (When Necessary)

If SHUTUP reaches this point, and is not the only job left running in the system apart from the EMT logger, SHUTUP makes one last attempt at killing all other remaining jobs. If SHUTUP is successful, it enters the next phase. If not successful, SHUTUP aborts operations with the following error message:

```
?SHUTUP failed in final job killing phase
??SHUTUP aborting - please try again later
```

4.2.10 EMT Logging Shutdown Phase (When Necessary)

The program enters the EMT logging shutdown phase if the EMT logging utility is present. (See Chapter 14 for a description of the EMT logger.) SHUTUP sends a shutdown message to the EMT logger, informing the EMT logger to stop. If the EMT logger does not shut down in 60 seconds, SHUTUP aborts the EMT logger with the following warning message:

```
%EMT logger failed to shutdown - killing EMT logger job = n
```

In this message, n is the job number of the EMT logger.

4.2.11 Remove Run-Time Systems and Resident Libraries Phase

During the remove run-time system and resident libraries phase, the program removes all run-time systems and resident libraries from the system tables. SHUTUP does not attempt to remove the primary run-time system.

4.2.12 Log File Closing Phase

During the log file closing phase, SHUTUP closes the log file, if it is open, and logs out. It then logs in again and goes on to the next phase.

4.2.13 Swap File Removal Phase

During the swap file removal phase, SHUTUP makes sure all installed swap files are removed.

4.2.14 Disk Dismount Phase

During the dismount phase, the program dismounts all mounted disks, both private and public, from the system. The system disk always remains mounted. If you have opened a file using the DCL commands OPEN or OPEN/LOG, or if you are executing SHUTUP from an indirect command file, and the file resides on a disk other than the system disk, an unexpected error occurs when SHUTUP tries to dismount the affected disk. SHUTUP aborts and you must close the file before retrying the shutdown.

4.2.15 Final Shutdown Phase

SHUTUP enters the final shutdown phase to output the last status message, clears the terminal buffers, and executes the special System shutdown SYS call (SYS -16). The monitor transfers control to the system initialization code (INIT.SYS) and bootstraps the system disk. The monitor loads the initialization code and INIT prints the Start Timesharing <Yes>? prompt (unless you are using automatic restart, in which case the system restarts).

4.2.16 SHUTUP Operation Examples

Each of the following sections contains an example of a SHUTUP operation. These examples and the description of the SHUTUP program should help you to shut down your system correctly and in orderly fashion.

4.2.16.1 SHUTUP Example: Large RSTS/E System

The system in the following example is being shut down in a leisurely fashion. This gives system and network users enough time to complete their tasks. SHUTUP allows users to log in to the system during the entire shutdown phase; logins are disabled at system shutdown, after 5 minutes. Those using the network can log in until the 5-minute mark, after that time the system does not allow network access by users. However, those on the network before new network activity is disabled can continue to work until the system shuts down completely.

```
$ run shutup
SHUTUP V9.7 RSTS V9.7 EDERE

10:11 AM 28-May-89  ########     Set-up Dialogue Phase       ########

Type '^' to any query to backup one question
```

```
Allow Print/Batch Services jobs to complete <YES>? yes
Use OPSER for utilities shutdown <YES>? yes
Allow OPSER utilities to reach logical end point <YES>? yes
Minutes until system shutdown (0-99) <5>? 15
Minutes until logins are disabled (0-99) <0>? 5
Minutes until new network activity is disabled (0-99) <5>? 5

Shutdown with automatic RESTART <NO>? NO


10:12 AM 28-May-89  ########     Warning Message Phase      ########
15 minute warning message sent
11 minute warning message sent
Further LOGINs are not disabled
Further network activity is now disabled
8 minute warning message sent
6 minute warning message sent
4 minute warning message sent
3 minute warning message sent
2 minute warning message sent
1 minute warning message sent

10:27 AM 28-May-89  ########     DECNET Shutdown Phase      ########

10:27 AM 28-May-89  ########   Initial Job Killing Phase    ########

10:27 AM 28-May-89  ########   Print/Batch Shutdown Phase   ########
Print/Batch Services shutdown complete at 10:27 AM

10:27 AM 28-May-89  ########       OPSER Shutdown Phase     ########
Detaching...

'OPSER' ATTACHING
#
JOB #6 'BA1SPL' TAKEN OFFLINE
#
JOB #7 'BA2SPL' TAKEN OFFLINE
#
JOB #5 'BA0SPL' TAKEN OFFLINE
#
JOB #4 'LP0SPL' TAKEN OFFLINE
#
JOB #3 'QUEMAN' TAKEN OFFLINE
#'OPSER' TERMINATING

Re-attaching...

10:29 AM 28-May-89  ########     EVTLOG Shutdown Phase      ########

10:29 AM 28-May-89  ########     ERRCPY Shutdown Phase      ########

10:29 AM 28-May-89  ########    Remove RTS/RES LIB Phase    ########

10:29 AM 28-May-89  ########     Log File Closing Phase     ########

10:29 AM 28-May-89  ########    SWAP File Removal Phase     ########

10:29 AM 28-May-89  ########      Disk DISMOUNT Phase       ########

10:29 AM 28-May-89  ########      Final Shutdown Phase      ########

Please wait for system to re-boot itself

RSTS V9.7 EDERE (DR1) INIT V9.7

28-May-89 10:30 AM

Start timesharing <Yes>?
```

4.2.16.2 SHUTUP Example: Small RSTS/E System

The following example shows a system shutdown when OPSER is not present. Except for the absence of the first OPSER-related question, the example applies when OPSER is present but not used during shutdown.

```
$ run shutup
SHUTUP V9.7 RSTS V9.7 EDERE

12:24 PM 22-May-89  ########     Set-up Dialogue Phase      ########

Type '^' to any query to backup one question

Allow Print/Batch Services jobs to complete <YES>? yes
Minutes until system shutdown (0-99) <5>?  5
Minutes until logins are disabled (0-99) <0>? 1

Shutdown with automatic RESTART <NO>? NO

12:24 PM 22-May-89  ########     Warning Message Phase      ########
5 minute warning message sent
Further LOGINs are now disabled
3 minute warning message sent
2 minute warning message sent
1 minute warning message sent

12:24 PM 22-May-89  ########   Initial Job Killing Phase    ########

12:25 PM 22-May-89  ########   Print/Batch Shutdown Phase   ########
Print/Batch Services shutdown complete at 12:25 PM

12:25 PM 22-May-89  ########     EVTLOG Shutdown Phase      ########

12:25 PM 22-May-89  ########     ERRCPY Shutdown Phase      ########

12:25 PM 22-May-89  ########    Remove RTS/RES LIB Phase    ########

12:25 PM 22-May-89  ########     Log File Closing Phase     ########

12:25 PM 22-May-89  ########    SWAP File Removal Phase     ########

12:25 PM 22-May-89  ########      Disk DISMOUNT Phase       ########

12:25 PM 22-May-89  ########      Final Shutdown Phase      ########

Please wait for system to re-boot itself

RSTS V9.7 EDERE (DR1) INIT V9.7

22-May-89 - 12:24 PM

Start timesharing <Yes>?
```

4.2.16.3 SHUTUP Example: DECnet/E Shutdown

This example shows SHUTUP when OPSER is used to shut down a spooling system of five spoolers. It also illustrates how the network shuts down when a system includes DECnet/E support and event logging by the DECnet/E EVTLOG program has been enabled.

```
$ run shutup
SHUTUP V9.7 RSTS V9.7 EDERE

12:39 PM 22-May-89  ########     Set-up Dialogue Phase      ########

Type '^' to any query to backup one question

Allow Print/Batch Services jobs to complete <YES>? yes
Use 'OPSER' for utilities shutdown <YES>? yes
Allow OPSER utilities to reach logical end point <YES>? yes
Minutes until system shutdown (0-99) <5>? 5
Minutes until logins are disabled (0-99) <0>? 0
Minutes until new network activity is disabled (0-99) <5>? 2

Shutdown with automatic RESTART <NO>? NO
```

```
12:40 PM 22-May-89  ########     Warning Message Phase     ########
Further LOGINs are now disabled
5 minute warning message sent

Further network activity is now disabled
3 minute warning message sent

Event type 2.0, Local node state change
Occurred 22-May-89 12:42:05.0 on node 135 (OTHG)
Reason for state change: Operator command
Old node state = On
New node state = Off

2 minute warning message sent
1 minute warning message sent

12:45 PM 22-May-89  ########     DECNET Shutdown Phase     ########

12:45 PM 22-May-89  ########   Initial Job Killing Phase   ########

12:45 PM 22-May-89  ########   Print/Batch Shutdown Phase   ########
Print/Batch Services shutdown complete at 12:45 PM

12:45 PM 22-May-89  ########       OPSER Shutdown Phase    ########
Detaching...

'OPSER' ATTACHING
#
JOB #6 'BA1SPL' TAKEN OFFLINE
#
JOB #7 'BA2SPL' TAKEN OFFLINE
#
JOB #5 'BA0SPL' TAKEN OFFLINE
#
MESSAGE            949 : 22-May-89 12:45 PM  JOB:3  DET  QUMRUN[1,2]
         BA1SPL (6) REQUESTED OFF-LINE -- TAKEN OFF-LINE
#
MESSAGE            950 : 22-May-89 12:45 PM  JOB:3  DET  QUMRUN[1,2]
         BA2SPL (7) REQUESTED OFF-LINE -- TAKEN OFF-LINE
#
MESSAGE            951 : 22-May-89 12:45 PM  JOB:3  DET  QUMRUN[1,2]
         BA0SPL (5) REQUESTED OFF-LINE -- TAKEN OFF-LINE
#
JOB #4 'LP0SPL' TAKEN OFFLINE
#
MESSAGE            952 : 22-May-89 12:45 PM  JOB:3  DET  QUMRUN[1,2]
         LP0SPL (4) REQUESTED OFF-LINE -- TAKEN OFF-LINE
#
JOB #3 'QUEMAN' TAKEN OFFLINE
#'OPSER' TERMINATING

Re-attaching...

12:45 PM 22-May-89  ########     EVTLOG Shutdown Phase     ########

Shutting down EVTLOG by operator request.

12:45 PM 22-May-89  ########     ERRCPY Shutdown Phase     ########

12:45 PM 22-May-89  ########    Final Job Killing Phase    ########

12:45 PM 22-May-89  ########    Remove RTS/RES LIB Phase   ########

12:45 AM 22-May-89  ########     Log File Closing Phase    ########

12:45 PM 22-May-89  ########    SWAP File Removal Phase    ########

12:45 PM 22-May-89  ########      Disk DISMOUNT Phase      ########

12:45 PM 22-May-89  ########      Final Shutdown Phase     ########
```

```
Please wait for system to re-boot itself

RSTS V9.7 EDERE (DR1) INIT V9.7

22-May-89 - 12:45 PM

Start timesharing <Yes>?
```

4.2.17 Notes on SHUTUP Operation

You can help the users on your system by establishing administrative procedures that govern the hours the system is in operation. If you have a fixed shutdown schedule, users can plan their work load and complete tasks in the allotted hours of timesharing.

You can keep them informed by placing messages in the system NOTICE.TXT file. The system displays the contents of NOTICE.TXT each time a user logs in using a comma (,) in the PPN. This is a convenient way to make sure all users know about important system operations.

Chapter ҁ

Managing Quotaꞩ

This chapter describes the quotas that limit the use of various system resources. You can assign quotas to a user when you create an account. RSTS/E quotas fall into four categories:

- Disk quotas
- Job quotas
- Send/Receive quotas
- System dynamic memory limits

The remainder of this chapter describes these quotas in detail. See Chapter 7 foı information on assigning or changing an account's quotas.

5.1 Disk Quotas

You can limit the amount of space available to individual users on public or private disks by creating and maintaining quotas on those disks. Quotas are maintained and enforced on a per-disk basis, except for the public structure where it is maintained across the structure.

You identify PPNs and assign quotas to them. The system automatically maintains usage counts during normal file activities.

Each account that exists on a Level 1.2 disk, except for the [0,1] account, has a disk quota attribute block associated with it. The quota blocks have the followinɛ information:

- Usage—Actual number of blocks on the disk taken up by your files
- Quotas—Maximum number of blocks on the disk that your files can take up before the system issues an error message

NOTE

For disk levels 1.1 and earlier, RSTS/E used only 16 bits to store the disk quotas. Thus, quotas could not be as large as the size of the largest disk supported by RSTS/E.

5.1.1 Disk Quota Operations

For pre-V9.0 disk structures, RSTS/E could only enforce disk quota limits at the time a user logged out. Thus, as a system manager, you could limit the amount of permanent disk space a user could have, but you could not limit the amount of temporary disk space.

On level 1.2 disks, RSTS/E maintains two types of disk related quotas: a logged-in quota and a logged-out quota. If at least one user is logged in to the file's account, the system automatically checks the usage count against the logged-in quota as users create, delete, and extend files. If no one is logged in to the file's account, then RSTS/E checks the usage count against the logged-out quota. The system also enforces the logged-out quota when a user logs out.

If extending the file causes the total disk usage to exceed the quota, RSTS/E rejects the operation and returns a ?Quota exceeded error message. When checking for a possible quota violation, RSTS/E takes into consideration the total write request. So, a write request that extends a file by multiple disk blocks is never only partially completed.

To perform the quota checks, the monitor keeps a record of the number of blocks that it allocates to each account on each disk. For public disks, this record is kept on a per disk basis, but, when quota checks are made, the counts are added up across the public structure. The monitor keeps the usage information in memory and writes it back to the disk when appropriate. If the system crashes (leaving old information on a disk), the disk rebuild operation (CLEAN) recomputes the usage and writes the correct data on the disk.

5.1.2 Logged-In Disk Quotas

Logged-in disk quotas are the number of disk blocks that the system lets a user allocate while logged in. The number of blocks in use is the total number of blocks allocated. The allocation of any file is its actual size in blocks, rounded up to the nearest full cluster. The allowable values are 0 to 16,777,214 (2^24—2), and UNLIMITED. Zero means that no disk storage is allowed. UNLIMITED is a special value that means the monitor does not enforce the logged-in quota.

Logged-in quotas do not apply on:

- Level 0 or Level 1.1 disk structures
- Disks mounted using MOUNT/NOQUOTA (although the usage is still kept up to date)
- Disks mounted using MOUNT/NOWRITE (because no file extends can take place on such a disk)
- DV0:, which is always mounted with /NOQUOTA

5.1.3 Logged-Out Disk Quotas

Logged-out disk quotas are the number of disk blocks that the system lets a user retain at logout time. The logged-out quota is also checked when a user tries to create or extend a file in a logged-out account.

The allocation of any file is its actual size in blocks, rounded up to the nearest full cluster. The number of blocks in use by any user is the total number of blocks allocated for all of the user's files.

The allowable values are 0 to 16,777,214 (2^24—2), or UNLIMITED. Zero means that no disk storage is allowed to remain allocated at logout time. UNLIMITED is a special value that means the monitor does not enforce the logged-out quota.

When a job logs out, the monitor checks the logged-out disk quota of each disk against the usage of that disk. For the public structure, the system disk quota is checked against the total usage on the public structure. If the quota is exceeded (and no other interactive jobs are logged in under the same account) then the monitor rejects the logout request.

For level 0 and level 1.1 disk structures, the logged-out quotas remain the same as pre-V9.0. The allowable values are 0 to 65,535, where zero means UNLIMITED.

5.2 Job Quotas

For pre-V9.0 disk structures, there was no restriction on the number of attached jobs that a user was allowed to run. Thus, a user who had access to several terminals could gain control of a large number of job slots. In V9.0, each account has a job quota.

The job quota specifies the total number of jobs (attached or detached) that each account can have active at any time. If the user attempts to exceed this quota, the system rejects the request and returns the ?Quota exceeded error message. If the user attempts to log in when the quota is reached, the system rejects the log-in request (see "Job Quotas and Login" below).

The default job quota is UNLIMITED.

5.2.1 Detached-Job Quota

The detached job quota is an account attribute. It indicates the actual number of jobs that are allowed to run detached under each account. If detaching a job causes this quota to be exceeded, then the system rejects the request and displays the following message:

```
?Quota exceeded
```

The quota check applies even if the user issuing the request is running under some other account. The detached job quota does not affect the automatic detaching operation that takes place when carrier is lost on a dial-up line controlling a job. In that case the job is unconditionally detached, but the quota is enforced when the user attempts to log back on (see the section "Job Quotas and Login").

The detached job quota defaults to zero when you create an account.

5.2.2 Job Quotas and Login

When a user attempts to log in, the system checks the current list of jobs under the account being logged in to against the quotas for that account. If either the detached job quota is exceeded or the job quota would be exceeded, the system rejects the log-in request. The system returns the list of currently detached jobs in the usual manner.

If a user attempts to log in to some account and the job quota has been reached, the result depends on whether there currently are jobs running detached under that account. If there are none, the system rejects the log-in request. If there are detached jobs, the system lets the user attach to one of them.

If a user attempts to log in to some account and the detached job quota for that account has been exceeded, the system rejects the request but the user can attach to one of the detached jobs. Thus, the monitor ensures that one user cannot create an unlimited number of detached jobs.

5.3 Send/Receive Quotas

You can also limit the number of receiver ID blocks (RIBs) and the number of messages that a job can have at any time.

5.3.1 RIB Quota

Specifies the maximum number of RIBs that a job running under an account can set up for itself as a message receiver. The allowable range is 0 to 256 or UNLIMITED. UNLIMITED is a special keyword that means no RIB quota is to be enforced.

5.3.2 Message Quota

Specifies the maximum total message quota that a job running under an account can request when declaring itself as a receiver. This maximum total must be shared by however many RIBs (local and network) the job wants to set up. The allowable range is 0 to 65,535 or UNLIMITED. UNLIMITED is a special keyword that means no message quota is to be enforced.

5.4 Dynamic Region Limits

Dynamic regions are areas in memory created by some application programs; Fortran 77 virtual arrays are examples of dynamic regions. You must have the INSTAL privilege to allocate dynamic regions. If users or applications do not have INSTAL, they can still call on a limited pool of publicly available space. The size of this pool is the dynamic region limit.

You can adjust the dynamic region limit with the DCL command SET SYSTEM/DYNAMIC_REGION_LIMIT. For example:

```
$ SET SYSTEM/DYNAMIC_REGION_LIMIT=20
```

This command sets the dynamic region limit at 20K words. As long as this command remains in effect, no user or application can take more than 20K words of dynamic region without the INSTAL privilege. There may be less than 20K words available at any given time, since some of the pool may be assigned to another user or application.

You need the SWCFG privilege to use the SET SYSTEM/DYNAMIC_REGION_LIMIT command.

Chapter 6

Managing Privileges

This chapter describes the privileges that you can assign to a user when creating an account. Privileges restrict the performance of certain system activities to certain users. These restrictions protect the integrity of the operating system's performance and thus the integrity of service provided to users. You should grant privileges to each user on the basis of two factors:

- Whether the user has a legitimate need for the privilege
- Whether the user has the skill and experience to use the privilege without disrupting the system

There are two types of privileges: RSTS/E-defined and user-defined.

6.1 RSTS/E-Defined Privileges

RSTS/E-defined privileges fall into several categories. The categories are based on the level of impact the user has on the system:

- User—Minimum privileges to use the system effectively
- Group—Potential to affect users inside the same group
- World—Potential to affect users outside the same group
- System—Potential to affect normal system operation
- File—Potential to read or write file data
- All—Potential to control the system

Users can use the DCL command SET JOB/PRIVILEGE to enable and disable privileges for which they are authorized. Table 6–1 lists the privileges by category and gives a brief definition.

The sections that follow Table 6–1 describe each privilege in alphabetical order. For a function-by-function description of privilege requirements, see the *RSTS/E Programming Manual*, the *RSTS/E System Directives Manual*, or the *RSTS/E System User's Guide*.

Table 6–1: RSTS/E Privileges

Category	Privilege	Activity Permitted
User	none	Most normal activity except those listed below
	SETPAS	Change own password
Group	GACNT	Management of all accounts in group
	GREAD	Read access to all files in group
	GWRITE	Write access to all files in group
World	EXQTA	Exceed quotas
	RDMEM	Read physical memory
	WACNT	Management of all accounts
	WREAD	Read access to all files
	WWRITE	Write access to all files
Files	DATES	Change file last access date
	DEVICE	Access restricted devices
	GREAD	Read access to all files in group
	GWRITE	Write access to all files in group
	INSTAL	Add/remove/list system files, run-time systems, libraries
	SYSIO	Set privilege bit (128) of a data file
	WREAD	Read access to all files
	WWRITE	Write access to all files
System	DATES	Change system clock
	HWCTL JOBCTL SWCTL	Perform system hardware control functions Perform job control functions Control logins and networking
	HWCFG SWCFG	Set device characteristics Set system parameters
	INSTAL	Add/remove/list CCL commands
	MOUNT	Mount and dismount devices
	PBSCTL	Control Print/Batch services
	SEND	Send to a restricted receiver, BROADCAST messages
	SHUTUP	Shut down the system
	TUNE	Alter priority, run burst, and swapping mode
All	RDNFS	Perform read-only non-file-structured I/O
	SYSIO	Write files in [0,*] (with WWRITE)
	SYSMOD	Modify the system
	TMPPRV	Set privilege bit of an executable program
	WRTNFS	Perform read/write non-file-structured I/O

6.1.1 DATES Privilege

The DATES privilege lets you change the system date or time. You also need thi privilege to change a file's date of last access or modification.

6.1.2 DEVICE Privilege

The DEVICE privilege lets you allocate, read, or write data on a restricted device

6.1.3 EXQTA Privilege

The exceed quota privilege (EXQTA) lets the space taken by the user's files on a given volume exceed any quota set for the user (as determined by the PPN) on that volume. However, this privilege does not let users exceed their logged-out quota when logging off the system.

This privilege also lets users specify a priority, page limit, CPU limit, or time limit value with a PRINT or SUBMIT command that exceeds the maximum valu for the specified queue.

The primary use of EXQTA is by privileged programs that need to perform certai operations regardless of user quotas. Normally, you would not assign EXQTA to users. Instead, you should set the user's quotas to get the effect you want.

6.1.4 GACNT Privilege

The group accounting privilege (GACNT) lets a user perform account management functions, such as create/delete account and change passwords/quotas for all accounts with the same project number (group) as the user. GACNT also lets a user create batch jobs in any account in the user's group.

6.1.5 GREAD Privilege

The group read privilege (GREAD) allows the user's job read and execute access to any file whose project number is the same as the user's.

6.1.6 GWRITE Privilege

The group write privilege (GWRITE) gives the user's job create, delete, rename, and write access to all files that have the same project number as the user's.

6.1.7 HWCFG Privilege

The hardware configuration privilege (HWCFG) lets you set terminal characteristics for any terminal, declare a device as restricted, create and delete the virtual disk, and set line printer parameters.

6.1.8 HWCTL Privilege

The hardware control privilege (HWCTL) lets you perform system control functions, for example seizing control of a device, or hanging up a modem line.

6.1.9 INSTAL Privilege

The install privilege (INSTAL) lets you add, remove, or list system files. You can also add or remove run-time systems, resident libraries, system logical names, or system commands (CCLs), and create dynamic regions above the dynamic region limit.

6.1.10 JOBCTL Privilege

The job control privilege (JOBCTL) lets you read the status of any other job, kill a job, or remove the receiver ID block (RIB) of another job. You can also create a job even if logins are disabled.

6.1.11 MOUNT Privilege

The MOUNT privilege lets you mount shared disks, or use certain restricted MOUNT command qualifiers.

6.1.12 PBSCTL Privilege

The PBSCTL privilege lets you control the Print/Batch Services. For example, you can start or stop servers, or change printer forms.

6.1.13 RDMEM Privilege

The RDMEM privilege lets you read memory (PEEK) and perform read-only mapping of physical memory by executing the .PLAS system directive.

6.1.14 RDNFS Privilege

The RDNFS privilege lets the user's job perform read-only non-file-structured I/O operations.

NOTE

> This privilege lets a job access data anywhere on the disk without the benefit of any file structure. Grant it only to users who need it. If this privilege is given to unqualified users who do not need it, the operating system and service to other users can be disrupted. Such disruptions can include the exposure of confidential information to unauthorized persons.

6.1.15 SEND Privilege

The SEND privilege lets you send a message to a restricted receiver. You also need this privilege to BROADCAST a message to a terminal.

6.1.16 SETPAS Privilege

The SETPAS privilege lets users change their own password. This privilege enables the SET PASSWORD command.

6.1.17 SHUTUP privilege

The SHUTUP privilege lets you run the SHUTUP program to shut down the system.

6.1.18 SWCFG Privilege

The SWCFG privilege lets you perform system configuration functions such as:

- Setting magnetic tape label defaults
- Setting date/time presentation formats
- Initializing queues and servers
- Managing DECnet

6.1.19 SWCTL Privilege

The software control privilege (SWCTL) lets you enable or disable logins. You also need this privilege to perform operational network management functions (for example, turning the network or individual circuits ON and OFF).

6.1.20 SYSIO Privilege

The SYSIO privilege lets the user's job:

- Create, modify, rename, or delete files in [0,*]
- Issue commands that write into files in [0,*] such as the DUMP/SYSTEM command
- Declare a network receiver that accepts inbound network connection requests (a network server program)
- Declare a receiver with a name that does not end with your job number
- Set the privilege bit in the protection code of a data file (execute bit (64) not set)
- Force commands to a terminal

6.1.21 SYSMOD Privilege

The SYSMOD privilege lets you perform functions that could modify the system:

- Write memory with POKE
- Map physical memory read/write
- Map the I/O page
- Perform non-file-structured reads and writes of a mounted disk

NOTE

This privilege lets a job access data anywhere on the disk without the benefit of any file structure. Grant it only to users who need it. If this privilege is given to unqualified users who do not need it, the operating system and service to other users can be disrupted. Such disruptions can include the destruction of information on the system device, the destruction of user data, and the exposure of confidential information to unauthorized persons.

6.1.22 TMPPRV Privilege

The TMPPRV privilege lets you set the privilege bit (128) in the protection code of a file with the execute bit (64) set. This privilege is not assigned to privileged programs when they are executed. Thus programs with temporary privilege cannot mark other programs as being privileged; a user must have TMPPRV privilege to perform this function.

NOTE

Grant the TMMPRV privilege with care because users with it can write and execute programs that perform system functions that are controlled by privileges the user does not have.

6.1.23 TUNE Privilege

The TUNE privilege lets you:

- Enable or disable disk caching
- Enable or disable file caching
- Lock your job in memory
- Set the maximum size any job can occupy in physical memory (SWAP MAX)
- Set the maximum time any job can run when compute bound (run burst)
- Set the priority of any job
- Change a batch server's priority and run burst

You can increase or decrease the priority of a job by executing the UU.PRI system directive or the DCL command SET JOB/PRIORITY. With the same privilege, a job can spawn another job with a priority other than its own. You can create such a job by using an optional argument to the system directive UU.JOB.

The TUNE privilege also lets you control whether a job can be swapped out of memory by executing the .SET system directive with the JFLOCK bit set.

Because of their significant effect on system performance, the BACKUP and RESTORE commands also require TUNE privilege.

NOTE

> Do not grant the TUNE privilege widely; if unqualified users have unrestricted ability to set job priorities, the fair and orderly scheduling of jobs for execution can be disrupted.
>
> Grant the TUNE privilege only to users who need to lock a job in memory for performance reasons. (Typically, this will be a real-time application.) If unqualified users have the unrestricted ability to lock jobs in memory, physical memory may be held unnecessarily, thus degrading system performance.

6.1.24 WACNT Privilege

The world accounting privilege (WACNT) lets you perform account management functions, such as create/delete account and change password/quotas, for all accounts. You can also use it to spawn jobs or run batch jobs in any account on the system.

To affect or to examine other accounts inside its own project number, a job needs only the GACNT privilege. But to affect or examine accounts outside its own project number, a job needs the WACNT privilege.

6.1.25 WREAD Privilege

The world read privilege (WREAD) gives you read and execute access to any file.

6.1.26 WRTNFS Privilege

The WRTNFS privilege lets you perform read/write non-file-structured I/O operations.

NOTE

> This privilege lets a job access data anywhere on the disk without the benefit of any file structure. Grant it only to users who need it. If this privilege is given to unqualified users who do not need it, the operating system and service to other users can be disrupted. Such disruptions can include the destruction of information on the system device, the destruction of user data, and the exposure of confidential information to unauthorized persons.

6.1.27 WWRITE Privilege

The world write privilege (WWRITE) gives you create, delete, rename, and write access to all files except those in [0,*]. For files in [0,*], you need to have SYSIO privilege as well as WWRITE privilege.

6.2 User-Defined Privileges

The user-defined privileges (USER1-8) are eight privileges that have been reserved for your own definition. Third-party and user software can test them at the system program level to control various aspects of their application programs, such as restricted functions or restricted access to certain files. RSTS/E does not use these privileges for any access or privilege checks.

6.3 Multiple Privileges and Jobs

The following sections describe how the monitor handles privilege information during the life of a job. They describe:

- Privilege Masks
- Job Creation
- Login
- Logout
- Spawned Jobs

6.3.1 Privilege Masks

A user's privileges are recorded in the job structure of the user's job in three privilege masks: authorized, current, and saved. When a user logs in to the system, the user's authorized and current privilege masks are initialized from the job's account attributes. The saved privilege mask is a copy of the current privilege mask upon entry to a privileged program. When temporary privileges are dropped, the current privilege mask is restored from the saved privilege mask.

6.3.2 Job Creation

At job creation, the monitor initializes both the current mask and the authorized mask, giving them all privileges except SYSMOD.

6.3.3 Login

When a job logs in, the system looks up the authorized mask in the account attributes. It copies this mask into the saved and authorized masks, ORs it into the current mask, and sets the job status to indicate the job has temporary privileges in effect.

If a program logs in, it now has all the privileges it originally had plus possibly some new ones. When the program exits, the user has all authorized privileges enabled; temporary privileges are dropped.

A user who logs in may not want all authorized privileges to be active at login. In that case the user can use a LOGIN.COM file to initially drop some privileges.

6.3.4 Logout

When a job logs out, the monitor clears the group-related privileges GACNT, GREAD, and GWRITE in all three privilege masks. This is done because the job is currently running with a project-programmer number (PPN) of [0,0] effectively putting it in group zero. The monitor drops group privileges because the intent oi these privileges is to allow access to the user's group, not group zero.

Apart from losing group privileges, a job neither gains nor loses any privileges as a result of logging out.

6.3.5 Spawned jobs

You can spawn a job using the UU.JOB system directive (see the *RSTS/E System Directives Manual*). A spawned job can be either logged-in or logged-out. For jobs spawned logged-in, the monitor usually gives the spawned job the set of authorized privileges for the account it logs in to. This is done before the program, if any, is run. If the program is a privileged program, the usual additional privilege processing takes place (see the section "Running a Privileged Program").

Jobs spawned logged-out are given the privileges of the job executing the UU.JOE directive.

Both logged-out and logged-in jobs spawned to an account other than the caller's require accounting privilege (GACNT/WACNT). Spawning therefore lets users with accounting privilege create jobs that have some other account's privileges, possibly more than their own.

6.4 Running Programs Under Multiple Privileges

The following sections explain how the monitor handles privileges when running programs. They describe:

- Running System Programs
- Temporary Privileges
- Designating a Program as Privileged
- Running a Privileged Program
- Program Exit

6.4.1 Running System Programs

System programs provide tools for performing common tasks, such as managing files and developing programs. Some programs are available to all users and require no particular privileges to run. They have a protection code <104>. EDT is an example of a nonprivileged system program.

Other system programs perform functions for privileged users. Most privileged system programs have a protection code <104>, unless world file access is required to successfully execute the program. In that case the protection code is <124>. The difference between the nonprivileged program and the privileged program that have the same protection code is that the latter verifies that a user has the required privileges before proceeding.

Some privileged system programs like SHUTUP have a protection code <232> but require the user to have a particular privilege to gain access. In the case of SHUTUP, the user needs the SHUTUP privilege to run the program. Access to privileged system programs depends on the user's set of privileges. When a user attempts to run a privileged program, the program first performs a privilege check to determine if the user is sufficiently privileged. If the user has the correct set of privileges, the program grants access. See the individual description of each system program to determine what privileges are required to gain access.

6.4.2 Temporary Privileges

A program that performs privileged functions for a nonprivileged user receives the special designation of privileged program. When run, it gives the user temporary privileges, that is, all privileges except SYSMOD and TMPPRV. When the program exits, the monitor withdraws the extra privileges from the user. Temporary privileges give the user access to privileged programs without compromising system security.

Sometimes programs need to perform privileged functions for nonprivileged users. For example, SHUTUP needs to PEEK at memory, an activity that requires RDMEM privilege. Because it is a program, SHUTUP can PEEK in a controlled manner. Therefore, it is not a security risk. On the other hand, unrestricted use of the RDMEM privilege by a user can compromise system security.

6.4.3 Designating a Program as Privileged

A user designates an executable program as privileged by setting the privilege bit (128) and the executable bit (64) in the protection code. Therefore, any program with a protection code of <192> or higher is privileged. The normal protection code associated with privileged executable programs is <232>, granting execute access to all, but restricting read/write access to the owner.

For security purposes, there are two restrictions regarding privileged programs:

- A user needs the TMPPRV privilege to be able to designate a program as privileged.
- A privileged program cannot reside on a disk that is mounted /NOSHARE. This restriction prevents an outsider from acquiring privileges by bringing in a privileged program on a private pack. To be able to mount a disk /SHARE, a user needs the MOUNT privilege. Thus, it is important that users who have the ability to mount disks /SHARE know the contents of the disk they are mounting.

6.4.4 Running a Privileged Program

When a user runs a privileged program, the monitor copies the account's current privilege mask into the saved privilege mask, then it ORs all privileges except SYSMOD and TMPPRV into the current privilege mask, giving the user temporary privileges (all but SYSMOD and TMPPRV).

It is good programming practice to have privileged programs drop privileges on entry, and raise them only when needed. This practice provides better protection against programming errors damaging the system. It thereby improves system security.

6.4.5 Program exit

Whenever a program exits or chains to another program, the monitor performs the following privilege-related cleanup:

- If temporary privileges are in effect, the monitor copies the saved mask back into the current mask. This cancels the temporary privileges.
- The monitor cancels any third-party privilege check currently in effect (see the *RSTS/E Programming Manual*, SYS call 31, or the *RSTS/E System Directives Manual*, general directive UU.3PP).
- If the job is currently logged out and does not have WACNT privilege, and the program exits, then the monitor kills the job. Passing control from one program to another program is possible without restriction when a job is logged out, but other operations that exit the current program result in a self-kill.
- If the program being exited is a privileged program, then the monitor clears the job's memory and sets the job size to the minimum size for the job's default keyboard monitor.

Chapter 7

Managing Accounts

This chapter gives background information on managing RSTS/E accounts and describes the DCL commands you use to create and maintain accounts. Table 7–1 summarizes these commands.

Table 7–1: DCL Commands for Managing Accounts

Command	Description
CREATE/ACCOUNT	Sets up a new account on a file-structured disk
DELETE/ACCOUNT	Deletes an account from a file-structured disk
SET ACCOUNT	Changes an account's attributes
SET PASSWORD	Changes an account's password
SHOW ACCOUNT	Displays account attributes

7.1 Accounts

An account is the structure by which RSTS/E recognizes a timesharing user, and in which RSTS/E maintains usage and access data. Each account is identified by a PPN, and can exist on any disk—public, or private. However, you can only log in to accounts that are defined on the booted system disk.

The public disk structure can be one or more disks and collectively these disk(s) are represented by the logical name, SY:. Within the public structure, the booted disk is known as SY0:. You can explicitly create, delete, or modify accounts on SY0: or any private disk, but not on other public disks. The system automatically extends accounts on public disks as needed. The only command that lets you specify a public disk in the device specification is SHOW ACCOUNT.

RSTS/E has two types of accounts:

- User—Contains files and login attributes
- Nonuser—Only contains files

The system decides which type account to create by the following rules:

- Explicitly by qualifier: /USER or /NOUSER
- Implicitly by the presence of any qualifier that suggests a user account such as /IN_QUOTA, /PRIVILEGES, /[NO]LOOKUP, or /[NO]NETWORK
- Neither of the above: /USER is assumed if the disk is SY0:, /NOUSER is assumed if the disk is private

7.1.1 Project-Programmer Number (PPN)

RSTS/E identifies each account by a project-programmer number (PPN). You can specify the project or programmer number as a single value or a range of values. You assign the PPN when you create the account. The PPN cannot be changed. Valid PPNs are: [0,1-254] and [1-254,0-254].

Note that [0,1-254] are legal PPNs for nonuser accounts only; they are not legal PPNs for user accounts. That is, no job can log in under [0,*]. Account [0,1] is created when the disk is initialized, and cannot be deleted. Also, [0,1-199] are reserved for DIGITAL software, [0,200] is reserved for RSTS/E and layered product installations, and [0,201-254] are reserved for user software packages.

7.1.2 Account Access Privileges

There are three categories of access to account information. A different set of privilege flags control each category. The categories are:

- Read information—Read accounting data and account attributes
  - None—You can read and display accounting information in the current account
  - GACNT—You can read and display accounting information of any account in your group
  - WACNT—You can read and display accounting information of any account
- Set information—Write accounting data and attributes to, or create or delete accounts
  - GACNT—You can write accounting data of any account in your group
  - WACNT—You can write accounting data of any account on the system
- Set password—Write (change) the passwords
  - SETPAS—You can change your own password (using the SET PASSWORD command only)
  - GACNT—You can change the password of any account in your group
  - WACNT—You can change the password of any account on the system

The privileges are inclusive: that is, if you have sufficient privilege to perform a given operation on any account in your group, you can also perform that operation on your own account. Similarly, if you have sufficient privilege to perform a given operation on any account on the system, you can perform that operation on any account in your group (including your own account).

7.2 Account Data

The following sections describe the data an account has to manage timesharing access and usage. Unless the section notes otherwise, each account element is:

- Initialized by CREATE/ACCOUNT
- Changed by SET ACCOUNT
- Displayed by SHOW ACCOUNT
- Deleted by DELETE/ACCOUNT

7.2.1 Account Name

The account name can be 1 to 13 characters, chosen from the ASCII character set whose decimal values are 9, 32 through 126, and 161 through 254, which includes all keyboard characters. This field is displayed by the SHOW ACCOUNT command, by PBS on job and file header pages, and by the BROADCAST command on message headers.

7.2.2 Date/Time/Keyboard of Last Login

The date and time that the account was most recently logged in to, and the keyboard at which that login occurred. The monitor maintains these data. You cannot change these fields with account management commands or monitor directives.

The keyboard number is useful only for hardwired terminals. Since LAT terminals are created as they are needed, their keyboard numbers are not useful identifiers. If you use OPSER logging, you can match up LAT keyboard numbers and login times to determine the server and port used.

7.2.3 Date/Time of Last Password Change

The date and time that the account's password was most recently changed. The monitor maintains these data. You cannot change these fields with account management commands or monitor directives.

7.2.4 Password

Normally, the system stores an account's password on the disk in such a way that it cannot be looked up. You select this form of storage with the /NOLOOKUP qualifier. If you need compatibility with earlier versions of RSTS/E, the /LOOKUP qualifier directs the system to store an account's password on the disk in such a way that the UU.RAD directive can be used to look it up (see the *RSTS/E System Directives Manual*). No DIGITAL-supplied program displays passwords.

For a /NOLOOKUP account, the password can be 6 to 13 characters long, chosen from the ASCII character set whose decimal values are 9, 32 to 62, 64 to 126, and 161 to 254, which includes all keyboard characters. For a /LOOKUP account, the password must consist of six alphanumeric characters.

A user can change an account's password using the SET PASSWORD command if the user has SETPAS privilege.

An account can also be set to the special state /NOPASSWORD_PROMPT. When this is so, LOGIN permits access to the account without requiring (or prompting for) a password. The password still exists; however, it is not used until the special state is cleared by setting /PASSWORD_PROMPT.

7.2.5 Interactive Flag

If set, the interactive flag lets a user log in under an account. If the interactive flag is not set, the system does not allow a job to be logged in under this account's PPN, except by the create job directive UU.JOB (see the *RSTS/E System Directives Manual*). No logins (by use of the UU.LIN system directive), interactive or otherwise, are allowed.

NOTE

Batch jobs are still allowed even if the interactive flag is not set.

7.2.6 Captive Account Flag

The captive account flag controls the way in which a job under a given account exits. If the captive flag is set, the system expects the job to remain totally under the control of a predetermined procedure, such as a login command procedure or a menu processor. Any attempt to exit to the keyboard monitor signals either an error or some devious action. The monitor converts such an exit request into a self-kill directive and removes the job.

7.2.7 Network Flag

The network flag, if set, lets you initiate a job under a given account over a network connection. This flag allows or disallows any network activity, for example, a remote COPY or SET HOST.

7.2.8 Dial-Up Flag

The dial-up flag, if set, lets you initiate a job under a given account over dial-up lines.

7.2.9 Expiration Date

The expiration date specified by this field is the last date on which you can initiate a job under the associated account. The system can be instructed (/EXPIRE without a date argument) never to allow initiation of a job under the PPN of this account. The system does not delete files after the expiration date. That is left to the system manager.

You can disable this field using /NOEXPIRE. The /NOEXPIRE qualifier lets an account remain usable indefinitely.

7.2.10 Authorized Privilege Mask

The authorized privilege mask specifies the maximum set of privileges that a job running under this account can exercise (other than by running privileged programs).

When a job logs in, all authorized privileges for the account are copied to the job's current privilege mask. You can use commands in your LOGIN.COM file to turn off some privileges at log in.

7.2.11 Creation Date

The date that the account was created. The system assigns the creation date when you create the account. The creation date cannot be changed.

7.2.12 UFD Cluster Size

UFD cluster size is the number of 512-byte blocks that each cluster allocated to the account's directory contains. You assign the UFD cluster size of a directory when you create an account. The UFD cluster size cannot be changed.

An account can have a maximum of seven clusters; therefore, the UFD clustersize affects the maximum number of files that can be stored in the account. This maximum value is further affected by the size of each file, its cluster size and attributes.

7.2.13 Quotas

You can assign quotas on disk usage, job slots, and message receiver slots when you create an account. See Chapter 5 for a discussion of these quotas.

7.2.14 System Data

The system account [0,1] on the system disk, is used to support certain system parameters such as the system password.

7.3 DCL Command Descriptions

The following sections describe the commands which you can use to manage accounts. All parameters can be specified on the command line.

7.3.1 CREATE/ACCOUNT Command

The CREATE/ACCOUNT command sets up a new account on a file-structured disk. CREATE/ACCOUNT requires GACNT or WACNT privileges.

Format

CREATE/ACCOUNT *dev:[p,pn]*

Command Qualifiers	Defaults
/[NO]CAPTIVE	/NOCAPTIVE
/CLUSTER_SIZE=ufd-cluster-size	none
/DETACHED_JOB_QUOTA=detached-job-quota	/DETACHED_JOB_QUOTA=0
/[NO]DIALUP	/DIALUP
/[NO]EXPIRE=date-spec	/NOEXPIRE
/IN_QUOTA=logged-in-quota	/IN_QUOTA=UNLIMITED
/[NO]INTERACTIVE	See discussion
/JOB_QUOTA=job-quota	/JOB_QUOTA=UNLIMITED
/[NO]LOG	/LOG
/[NO]LOOKUP	/NOLOOKUP
/MESSAGE_QUOTA=message-quota	/MESSAGE_QUOTA=12
/NAME="account name"	none
/[NO]NETWORK	/NETWORK
/OUT_QUOTA=logged-out-quota	/OUT_QUOTA=UNLIMITED
/[NO]PASSWORD_PROMPT	/PASSWORD_PROMPT
/POSITION=position	/POSITION=INDEX
/PRIVILEGES=(priv,...)	/PRIVILEGES=SETPAS
/RIB_QUOTA=rib-quota	/RIB_QUOTA=3
/SIZE=ufd-size	/SIZE=1
/TEMPLATE=dev:[p,pn]	none
/[NO]USER	See discussion

Prompts	Defaults
Account:	
Password:	Optionally, for user accounts
Password again, for verification:	Optionally, for user accounts

Command Parameters

dev:[p,pn]

Device specification of a mounted device and the PPN for the account to create. You must specify either the device specification or PPN. The default value for dev: is _SY0:. The default PPN is your PPN. You can have wildcards and ranges in the PPN specification. The allowable values for the PPN depend on the privileges you have.

Command Qualifiers

/[NO]CAPTIVE

Indicates whether the account is to be a captive account. The default is /NOCAPTIVE.

/CLUSTER_SIZE=ufd-cluster-size
The number of 512-byte blocks that each cluster in this account's UFD contains. Acceptable values are 1, 2, 4, 8, or 16. Disks with a PCS of 32 or 64 require a cluster size of 16. The value cannot be less than the pack cluster size of the specified disk.

/DETACHED_JOB_QUOTA=*detached-job-quota*
See Chapter 5. The default is 0.

/[NO]DIALUP
Indicates whether logins from dial-up terminal lines are allowed. The default is /DIALUP.

/[NO]EXPIRE=date-spec
/NOEXPIRE means that the account is never to expire.

/EXPIRE=date-spec means that the account expires at the end of the date you specify.

/EXPIRE without a date specification means that the account is to be marked as "already expired," that is, unavailable for any job to run or be created under its PPN. The default is /NOEXPIRE.

/IN_QUOTA=logged-in-quota
See Chapter 5. The allowable values are 0 to 16,777,214 and UNLIMITED. Zero means that no disk storage is allowed. UNLIMITED is a special keyword that means the monitor does not enforce the logged-in quota. The default is UNLIMITED.

/[NO]INTERACTIVE
This qualifier controls the way in which you can create a job under this account. If you set /NOINTERACTIVE, the system does not allow jobs to be logged in under this account's PPN except by the create job directive UU.JOB (see the *RSTS/E System Directives Manual).*

The default is /INTERACTIVE.

/JOB_QUOTA=job-quota
See Chapter 5. The default is UNLIMITED.

/[NO]LOG
If /LOG is in effect, the CREATE function logs the creation of the new user or nonuser account with one of the following messages:

```
User account dev:[p,pn] created at <device cluster number>
Non-user account dev:[p,pn] created at <device cluster number>
```

The default is /LOG.

/[NO]LOOKUP
/NOLOOKUP means that RSTS/E stores the password in such a way that it cannot be looked up. The passwords must be six to fourteen characters long.

/LOOKUP means that RSTS/E stores the password in a manner that follows the rules for RSTS/E versions earlier than V9.0. The passwords must be six characters long.

The default is /NOLOOKUP.

/MESSAGE_QUOTA=message-quota

See Chapter 5. The allowable values can be from 0 to 65,535 or UNLIMITED. The UNLIMITED keyword means that the monitor does not enforce the message quota.

The default is 12.

/NAME="account name"

Can be 1 to 13 characters, chosen from the ASCII character set whose decimal values are 9, 32 to 126, and 161 to 254, which includes all the keyboard characters.

/[NO]NETWORK

Indicates whether logins over network connections are allowed.

The default is /NETWORK.

/OUT_QUOTA=logged-out-quota

See Chapter 5. The allowable values are 0 to 16,777,214 and UNLIMITED. Zero means that no disk storage is allowed to remain at logout time. UNLIMITED is a special keyword that means the monitor does not enforce the logged-out quota.

The default is UNLIMITED.

/[NO]PASSWORD_PROMPT

/NOPASSWORD_PROMPT means that LOGIN is not to require or prompt for a password for a user to gain access to the specified account (that is, specifying the [p,pn] in response to User: is sufficient to log a user in to the account).

The default is /PASSWORD_PROMPT.

/POSITION=position

The starting position of the account's UFD expressed in device cluster numbers (DCNs). Legal values are 1 through the maximum DCN for the specified disk.

MIDDLE is a special keyword that means create the UFD for this account as near as possible to the middle of the disk. INDEX is a keyword value that means create the UFD as near as possible to the storage allocation table (file [0,1]SATT.SYS).

The default is INDEX.

/PRIVILEGE=(priv,...)

The possible values for each priv are: NONE, ALL, a privilege keyword, or a privilege keyword with a NO prefix. ALL means all privileges that the current user has enabled.

RSTS/E scans the list from left to right, with each successive term adding or taking away privileges. For example, (ALL,NOWWRITE) means all privileges except WWRITE.

If you specify /TEMPLATE=dev:[p,pn], RSTS/E treats the template account's privileges as though they appeared to the left of all the entries in the /PRIVILEGE qualifier.

/RIB_QUOTA=rib-quota

See Chapter 5. The allowable values can be from 0 to 256 or UNLIMITED. The UNLIMITED keyword means that the monitor does not enforce the RIB quota.

The default is 3.

/SIZE=ufd-size

The initial size of the new account's UFD expressed in clusters. Acceptable values are 0 - 7.

The default is 1.

/TEMPLATE=dev:[p,pn]

If present, this qualifier indicates that the specified account's parameters are to be used as defaults for the new account's parameters. You must specify either the device specification or the PPN.

The default value for dev: is _SY0:. The default PPN is your PPN.

NOTE

> The specified account's password, size, and position are not part of the defaults for the new account.

/[NO]USER

Indicates the type of account to create. /USER means create an account with login attributes and file storage capabilities. /NOUSER means create an account for file storage only.

The default is /NOUSER.

7.3.2 DELETE/ACCOUNT Command

The DELETE/ACCOUNT command removes an account from a file-structured disk. DELETE/ACCOUNT requires WACNT or GACNT privilege and WWRITE or GWRITE privilege. You also need SYSIO privilege to delete a [0,*] account.

Format

DELETE/ACCOUNT *dev:[p,pn]*

Command Qualifiers	Defaults
/[NO]LOG	/LOG
/[NO]PURGE[=QUERY]	See discussion
/[NO]QUERY	See discussion
/[NO]RESET[=QUERY]	See discussion
/[NO]USER	See discussion

Prompts

Account:

Command Parameters

dev:[p,pn]

Device specification of a mounted disk, and the PPN of the account to delete from the specified disk. You can omit the device specification (the default is _SY0:); you must specify the PPN. The PPN field allows wildcards and ranges. Unless you specify /NOQUERY, the system requests confirmation that you really want to delete the account:

```
Really delete account dev:[p,pn] <NO> ?
```

Command Qualifiers

/[NO]LOG

If /LOG is in effect, the DELETE function logs the deletion of the account:

```
Account dev:[p,pn] deleted
```

The default is /LOG.

/[NO]PURGE[=QUERY]

This qualifier controls the actions to be taken if the account contains files. When you specify /PURGE, DELETE/ACCOUNT deletes any files which it finds, and then deletes the account. If you specify the optional argument (=QUERY), RSTS/E displays a warning message if files exist for the specified account and then requests permission to proceed.

When you specify /NOPURGE, DELETE/ACCOUNT displays a warning message if files exist in the account, skips the account, and goes to the next account if a wildcard account was specified:

```
%Account dev:[p,pn] has n blocks of disk allocated
Account dev:[p,pn] not deleted
```

If you do not specify /PURGE, /NOPURGE, or if /PURGE=QUERY is specified, DELETE/ACCOUNT proceeds if no files exist. Otherwise, before proceeding, RSTS/E displays a prompt indicating that there are files in the account:

```
Proceed (Y/N) <N> ?
```

Deleting files (either by /PURGE or by answering YES to the prompt) requires GWRITE or WWRITE privilege. If any file deleted by DELETE/ACCOUNT has protected data (protection code = 128), the system writes over the file with zeros before deletion.

/[NO]QUERY

Controls the action taken if the account has nonzero accounting data or contains files.

/QUERY indicates that RSTS/E displays a prompt before deleting any account. /NOQUERY indicates that RSTS/E only displays a message if the account has nonzero accounting data or contains files.

The default is /NOQUERY.

/[NO]RESET[=QUERY]

Controls the action taken if the specified account has nonzero accounting data (CPU time, device time, connect time, or kilo-core-ticks). If you specify /RESET, RSTS/E deletes the account even if it has nonzero accounting data.

If you specify the optional argument (=QUERY), RSTS/E displays a warning message that the specified account has nonzero accounting data and asks for permission to proceed.

If you specify /NORESET, DELETE/ACCOUNT displays a warning message if the account has nonzero accounting data, skips the account, and goes to the next account if a wildcard account was specified:

```
%Account dev:[p,pn] has nonzero accounting statistics
Account dev:[p,pn] not deleted
```

If you do not specify /RESET, /NORESET, or if /RESET=QUERY is specified, the delete account operation proceeds if the accounting data is zero. Otherwise, RSTS/E displays a warning message indicating that the specified account has nonzero data and asks for permission to proceed.

/[NO]USER

You can restrict the search for a given account to a specific type (user or nonuser) by using this qualifier. The default is to search all accounts.

Examples

1.
```
$ DELETE/ACCOUNT/PURGE/RESET/NOQUERY
```

   This command allows you to delete an account regardless of its current state.

2.
```
$ DELETE/ACCOUNT/NOPURGE/NORESET
```

   This command prevents you from deleting an account that has files or nonzero accounting data.

3.
```
$ DELETE/ACCOUNT/QUERY
```

   This command allows you to be prompted before deleting an account.

4.
```
$ DELETE/ACCOUNT
```

   This command allows you to be prompted only for an account that has files or nonzero accounting data.

7.3.3 SET ACCOUNT Command

The SET ACCOUNT command modifies account parameters on a file-structured disk. SET ACCOUNT requires GACNT or WACNT privilege.

Format

SET ACCOUNT *dev:[p,pn]*

Command Qualifiers	Defaults
/[NO]CAPTIVE	none
/DETACHED_JOB_QUOTA=detached-job-quota	none
/[NO]DIALUP	none
/[NO]EXPIRE[=date-spec]	none
/IN_QUOTA=logged-in-quota	none
/[NO]INTERACTIVE	none
/JOB_QUOTA=job-quota	none
/[NO]LOG	/LOG
/[NO]LOOKUP	none
/MESSAGE_QUOTA=message-quota	none
/NAME="account name"	none
/[NO]NETWORK	none
/OUT_QUOTA=logged-out-quota	none
/[NO]PASSWORD_PROMPT	none
/PRIVILEGES=(priv,...)	none
/RIB_QUOTA=rib-quota	none
/[NO]USER	none

Prompts

Account:

Command Parameters

dev:[p,pn]

Device specification of a mounted disk and the PPN of the account to modify on the specified disk. You can omit either the device specification or the PPN, but not both. The default device specification is _SY0:. The default PPN is the user's PPN. You can have wildcards and ranges in the PPN specification. RSTS/E uses all qualifiers that apply to a specific account and displays the following informational message:

```
Account dev:[ppn] modified
```

If no qualifiers apply to a selected account, the system displays the following warning message:

```
%Account dev:[ppn] not modified
```

Command Qualifiers

/[NO]CAPTIVE

Indicates whether the account is to be a captive account.

/DETACHED_JOB_QUOTA=*detached-job-quota*

See Chapter 5.

/[NO]DIALUP

Indicates whether logins from dial-up terminal lines are allowed.

/[NO]EXPIRE=date-spec

/NOEXPIRE means that the account is never to expire.

/EXPIRE=date-spec means that the account is to expire at the end of the day specified.

/EXPIRE without a date specification means that the account is to be marked as unconditionally expired, that is, unavailable for any job to run under its PPN.

/IN_QUOTA=logged-in-quota

See Chapter 5.

/[NO]INTERACTIVE

This qualifier controls the way in which you can create a job under this account. If you set /NOINTERACTIVE, the system does not allow jobs to be logged in under this account's PPN except by the create job directive UU.JOB (see *RSTS/E System Directives Manual*).

/JOB_QUOTA=*job-quota*

See Chapter 5.

/[NO]LOG

If /LOG is in effect, the SET ACCOUNT function logs the modification of the account:

```
Account dev:[p,pn] modified
```

If /NOLOG is in effect, only warning messages are logged.

The default is /LOG.

/[NO]LOOKUP

/NOLOOKUP means that RSTS/E stores the password in such a way that it cannot be looked up. /LOOKUP means that RSTS/E stores the password in a manner that follows the pre-V9.0 rules.

/MESSAGE_QUOTA=*message-quota*

See Chapter 5.

/NAME="account name"

Can be 1 to 13 characters, chosen from the ASCII character set whose decimal values are 9, 32 to 126, and 161 to 254.

/[NO]NETWORK

Indicates whether logins over network connections are allowed.

/OUT_QUOTA=logged-out-quota

See Chapter 5.

/[NO]PASSWORD_PROMPT

If you specify /NOPASSWORD_PROMPT, the account is set to not require a password for logging in. To resume password prompting, specify /PASSWORD_PROMPT; the password is the same as it was before /NOPASSWORD_PROMPT was specified.

/PRIVILEGE=(priv,...)

This qualifier modifies the privileges currently assigned to the account. A priv specification occurring in the positive form adds the associated privilege to the account's authorized privilege mask; a NOpriv specification removes the associated privilege. Note that the change affects only future logins; jobs already logged in are unaffected.

A positive specification (priv) is effective only if you are authorized to have that privilege. However, any user with WACNT or GACNT (as appropriate) can remove any privilege.

/RIB_QUOTA=*rib-quota*

See Chapter 5.

/[NO]USER

You can restrict the search for a given account to a specific type (user or nonuser) with this qualifier.

The default is to search all accounts.

7.3.4 SET PASSWORD Command

The SET PASSWORD command lets you change the password of the specified user account. When you issue this command, the system responds with:

```
New password:
```

You then enter the desired new password. If the entry is invalid or a null string (Return only), the system gives the error message ?Invalid password, and returns to the New password prompt. The system accepts an oversize password string; however, it ignores the excess characters and displays a warning message.

After you enter a new password, the system prompts:

```
New password again, for verification:
```

You then reenter the same password. If the passwords do not match, the system displays the error message ?Verification mismatch, does not change the password, and returns to command level. If the passwords match, the system displays the /LOG message, if any, and returns to command level.

Users need SETPAS (or either GACNT or WACNT) privilege to change their own password. Users need GACNT or WACNT privilege to change another account's password. If you only have SETPAS, the system prompts you for the old password:

```
Old password:
```

You must enter the correct current password for your account. A mismatch results in the error message:

```
?Old password validation error.
```

Format

SET PASSWORD *[dev:[p,pm]]*

Command Qualifiers	**Defaults**
/[NO]LOG	/LOG
/SYSTEM	none

Prompts

See description

Command Parameters

dev:[p,pn]

Device specification of a mounted disk, and the PPN of the account for which to change the password. You can omit the device specification, the PPN, or both. The default device specification is _SY0:. The default PPN is the user's PPN. If a nonuser account is specified, the system displays a warning message:

```
% [ppn] is not a user account
```

You can have wildcards and ranges in the PPN specification. If a wildcard or range specification includes a nonuser account, RSTS/E skips the account and does not display a message.

NOTE

> Users who only have SETPAS privilege cannot specify a dev:ppn parameter; in that case, RSTS/E rejects the command and displays the following error message:

```
?Need xACNT privilege to access dev:[ppn]
```

Command Qualifiers

/[NO]LOG

If you specify /LOG, successful setting of a new password yields:

```
Password set for account dev:[p,pn]
```

The default is /LOG.

/SYSTEM

If you specify /SYSTEM, the new password replaces the old system password. You must have WACNT privilege to change the system password.

7.3.5 SHOW ACCOUNT Command

The SHOW ACCOUNT command displays account attributes. No privilege is required to display your own account. You need GACNT or WACNT privilege to display other accounts.

The following are samples of the output from the SHOW ACCOUNT command with different qualifiers.

```
$ SHOW ACCOUNT/BRIEF [*,194]

Account  Name         Allocation  IDNCLP Last Login                Expires

[1,194]  Blade Runner       6016  IDN  P KB26:  03-Dec-89 01:24 PM 15-AUG-90
[2,194]  JT Sysgen          5080  I    P KB32:  12-Nov-89 02:56 PM
[3,194]  JT PIP             4336  IDN  P KB32:  06-Nov-89 03:06 PM
[4,194]  JT ACTMGR          3280  IDN  P
[10,194]                     752

5 accounts matching _SY0:[*,194]
Total allocation:      19464
```

NOTE

If an asterisk (*) character appears after the value in the allocation field, it means the account is over quota.

```
$ SHOW ACCOUNT/FULL

_SY0:[1,214]  "Blade Runner"                  Created 09-May-90

   Privileges:  GACNT  WACNT  GREAD  GWRITE WREAD  WWRITE DATES  DEVICE EXQTA
                HWCFG  HWCTL  INSTAL JOBCTL MOUNT  PBSCTL RDMEM  USER0  USER1
                USER2  USER3  USER4  USER5  USER6  USER7  RDNFS  SEND   SETPAS
                SHUTUP SWCFG  SWCTL  SYSIO  SYSMOD TUNE   WRTNFS

   Attributes:  INTERACTIVE   DIALUP     NETWORK    NOCAPTIVE
                NOLOOKUP      PASSWORD_PROMPT       EXPIRES: 15-Aug-90

   Quotas:  Disk usage   - Logged out:  unlimited   Logged in:    unlimited
            Job limits   -   Detached:  unlimited       Total:            3
            Send/Receive -        RIB:        256     Message:    unlimited

   Accounting:      CPU Time:   58:23:24.9    Kilo-Core-Ticks:     52377442
                 Device Time:  214:35         UFD Clustersize:           16
                Connect Time:  903:25        Blocks allocated:         6000

                Last Password Change on 28-Nov-89 at 03:21 PM
                Last login on _KB26: on 03-Dec-89 at 01:24 PM
```

NOTE

If an asterisk (*) appears after the value in any field in the quotas section, it means the account is over quota.

```
$ SHOW ACCOUNT/ACCOUNTING_DATA [*,194]
Account   Name          Allocation        KCT      CPU-time   Connect    Device

[1,194]   Blade Runner        6016   52377442    58:23:24.9    903:25    214:35
[2,194]   JT Sysgen           5080       2215    00:00:17.5     00:15     00:00
[3,194]   JT PIP              4336        859    00:00:08.6     00:17     00:00
[4,194]   JT ACTMGR           3280        797    00:00:09.7     00:02     00:00
[10,194]                       752

5 accounts matching _SY0:[*,194]
Total allocation:      19464
```

NOTE

If an asterisk (*) character appears after the value in the allocation field, it means the account is over quota.

Format

SHOW ACCOUNT *dev:[p,pn]*

Command Qualifiers	Defaults
/ACCOUNTING_DATA	See discussion
/ALL	See discussion
/BRIEF	/BRIEF
/FULL	none
/OUTPUT=filespec	none
/[NO]RESET	/NORESET
/[NO]USER	See discussion

Prompts

none

Command Parameters

dev:[p,pn]

Device specification of a mounted disk, and a PPN. The PPN can be a wildcard or range specification (for appropriately privileged users). You can omit both the device specification and the PPN. The default device specification is _SY0:. The default PPN is the user's PPN.

Command Qualifiers

/ACCOUNTING_DATA

Produces a one-line display of accounting data for the specified accounts. This qualifier conflicts with /BRIEF and /FULL. If you specify them in combination, the rightmost qualifier overrides the others and execution proceeds without an error.

/ALL

Produces a display of all accounts on the specified disk. If an account is specified along with /ALL, the account specification overrides /ALL.

/BRIEF

Produces a display that uses only one line per account. You cannot use /BRIEF in combination with /RESET. When shown the display, the letters IDNCLP indicate that the account has the following attributes:

```
I = Interactive
D = Dial-up
N = Network
C = Captive
L = Lookup password
P = Password prompt
```

/FULL

Produces a complete display for the accounts specified.

/OUTPUT=filespec

If specified, the display is written to the file you specify.

/[NO]USER

Specifies which accounts are to be included in the display. /USER means only display accounts with login attributes. /NOUSER means only display accounts without login attributes.

The default is to display all accounts.

/[NO]RESET

Indicates whether the accounting data is reset after being displayed. The /RESET qualifier requires GACNT or WACNT privilege. You cannot use /RESET in combination with /BRIEF. If neither /ACCOUNTING_DATA or /FULL is specified, /ACCOUNTING_DATA is assumed.

The default is /NORESET.

Chapter 8

Managing Terminals

This chapter gives background information on managing terminals, either on your system or connected by Local Area Transport (LAT). The RSTS/E operating system supports a variety of terminals. The RSTS/E operating system automatically sets the default characteristics of all interfaces to certain standard values:

- Produces hard-copy output
- Prints data up to 80 columns wide
- Sets autobaud detection on multiplexed lines

You must use the SET TERMINAL command to set the characteristics of the terminals that do not have these standard default values.

8.1 Understanding the SET TERMINAL Command

The SET TERMINAL command sets characteristics for terminals attached locally to a RSTS/E system or for terminals connected by dial-up lines. Users can use the SET TERMINAL command to set characteristics for their own terminals. Only users with HWCFG privilege can use the SET TERMINAL command to define characteristics for other terminals.

The /AUTOBAUD qualifier lets the system detect and set the baud rate of any multiplexed terminal when the user logs in. As system manager, you can include SET TERMINAL/NOAUTO/PERM commands in the START.COM file for all multiplexed terminals you want to manually set the baud rate.

There are two methods for setting the characteristics of dial-up lines:

- Nonprivileged user sets characteristics. All dial-up lines start out with the standard default settings. Consequently, the user of a dial-up line must log in to the system at the standard default settings, and then use the SET TERMINAL command to set the characteristics of the terminal. Note that the terminal characteristics revert to the standard settings when the user logs out.
- A user with HWCFG privilege sets permanent characteristics. You can issue the SET TERMINAL/PERMANENT command to set the permanent characteristics. This causes the system to automatically use the characteristics you set, instead of the standard values, each time a user logs in on a particular dial-up line. Note that the permanent characteristics remain in effect for the current time-sharing session, unless you issue a new SET TERMINAL command.

8.2 Terminal Line Speed Characteristics File: TERSPD.SYS

You need to create a special terminal line speed characteristics file if you have these two circumstances:

- You decide to restrict the valid line speeds at certain terminals to a subset of the line speeds allowed by the terminal interface.
- You install an interface that has been modified so nonstandard line speeds are substituted for one or more standard interface line speeds.

The following guidelines can help you prepare a terminal line speed characteristics file:

1. Gather the information on the restrictions or modifications and the keyboard numbers of the terminals affected.
2. Create a file named TERSPD.SYS with a text editor.
3. Include all of the information from step 1 in the file.
4. Place the file in the system library account.

NOTE

> The /AUTOBAUD qualifier does not honor the restricted speeds in TERSPD.SYS.

You can use the file TERSPD.SYS to allow only certain speeds on certain terminals. The format for each line of the file is the terminal device designator, a colon, and a list of speeds to allow for that terminal; each speed separated by a comma. For example:

```
KBF2: 300,1200,...
```

The terminal device designator can be KBnn:, TTnn:, or KBcnn:. The form KBnn: or TTnn: produces a result that may change as new hardware is added. In the form (KBcnn:), c indicates the hardware controller and nn is a subline on that type controller. Table 8–1 lists controller types that RSTS/E supports.

DIGITAL recommends that you use the controller form to specify terminal devices in the START.COM, TERSPD.SYS, and TERDFL.SYS files. In this way, if you change the number of pseudo keyboards, you do not have to change those files. The SHOW DEVICE command displays both forms of the device designator for each keyboard.

Table 8–1: RSTS/E Supported Controller Types

Code	Controller Type
B	DL11 C,D type single line interface
C	DL11E type single line modem interface
D	Pseudo keyboards
F	DH multiplexor
G	DZ/DZV/DZQ multiplexor
H	DHV/DHU/DHQ/DHF/CXF/CXA/CXB multiplexor

You can specify up to 16 speeds for each terminal. You can specify a range of terminals on one line as low-end terminal device designator, an optional colon, dash, high-end terminal device designator, mandatory colon, and then the list of speeds.

For example:

```
KBG0-KBH8: 300,1200,...
```

When you specify a terminal number without any speeds after it, no speeds are allowed and the speed cannot be changed. You can use blank lines, comment lines (lines beginning with an exclamation mark), and comments at the end of lines to document the file.

An example of a TERSPD.SYS file follows:

```
!
!      T E R S P D . S Y S
!
!      This file is used by the system manager to define
!      those speeds which are allowed on each terminal.
!      The format is as follows:
!      KBn:list of speeds separated by commas.
!
KBF0:                   !No speeds allowed for KBF0
KBF1:2400,9600          !KBF1 can have either 2400 or 9600
KBF2:-KBF4:9600         !KBF2 to KBF4 can have 9600 only
KBF5:9600,EXTA,EXTB     !KBF5 can use 9600 & external speeds
KBF6:0,50,75,110,150,200,300,600,1200,1800,2400,4800,9600
KBF7:0,50,75,110,150,200,300,600,1200,1800,2400,4800,9600,EXTA,EXTB
KBF8-KBF15:0,50,75,110,150,200,300,600,1200,1800,2400,9600
KBG0-KBH8:50,300,1200,2400,4800,9600
KBH9:0,50,75,110,150,200,300,600,1200,1800,2400,9600
KBH10:-KBH15:300,600,1200 !KBH10 to KBH15 are dial-up lines
```

NOTE

> The DL11E-type interface and the individual local interfaces (KL11 and DL11A through DL11D) do not have programmable baud rates. For this reason, you cannot execute commands to change baud rates on a keyboard line having any of those interfaces (KBAnn, KBBnn, and KBCnn).

8.3 Setting Terminals Other Than Your Own

If you have HWCFG privilege, you can set the characteristics of other terminals in the RSTS/E system with the SET TERMINAL command. For example, if you want to set the characteristics of a VT100 terminal at KB32, type:

```
$ SET TERMINAL/VT100 KB32:
$
```

After you type the keyboard number and press the Return key, RSTS/E immediately sets the characteristics of the line to those of a VT100 alphanumeric display terminal. DCL prints the dollar sign ($) prompt again to let you enter another command.

You can also change specific characteristics of a terminal. For example, to limit the line length of the terminal at KB40, type:

```
$ SET TERMINAL/WIDTH=60 KB40:
$
```

Each time 60 characters print on KB40, the system performs a carriage return and line feed.

Use the /PERMANENT qualifier to specify characteristics on a logged-out terminal. For example, you can use /PERMANENT to set an initial WIDTH characteristic for a pseudo keyboard:

```
$ SET TERMINAL/WIDTH=70/PERMANENT KBD0:
```

8.4 User-Defined Terminal Characteristics File: TERMAC.SYS

Each line of TERMAC.SYS contains the unique name for a terminal type and the command qualifiers that define the characteristics of that terminal type. The qualifiers are explained in the description of the SET TERMINAL command at the end of this chapter.

The /TYPE=n qualifier assigns a unique 8-bit binary value to each terminal type. The values 0 to 128. are reserved for use by DIGITAL. The values 129. to 255. are available for user-defined terminal types. The SHOW TERMINAL command uses this value to locate a terminal's characteristics.

An example of a TERMAC.SYS file follows:

```
!
!       T E R M A C . S Y S
!
VT125X/DEV=VT125/WIDTH=80/SPEED=(300,9600)/TYPE=201
GIGI/DEV=VK100/SETUP=GREEN/TYPE=202   ! VK100 W/GREEN BACKGROUND
DIABLO/DEV=LA34/SPEED=9600/WID=80/SETUP=DIABLO/TYPE=203
LA34X/DEV=LA34/TAB/SPEED=9600/TYPE=204
VT52X/DEV=VT52/SPEED=1200/TYPE=205
FOO/DEV=BAR/WIDTH=132/TYPE=206
BAR/DEV=LA120/SETUP=NARROW/WIDTH=80/TYPE=207
```

For example, to define an LA120 that has only 80-column paper, place the following line in the file:

```
MYLA120/DEVICE=LA120/WIDTH=80/TYPE=208
```

Then, use the following command to set KB19: to these characteristics:

```
$ SET TERMINAL/DEVICE=MYLA120/PERMANENT KB19:
```

You can initialize your terminals to the known terminal types and their default characteristics through the use of the TERMAC.SYS file. You can use blank lines, comment lines (lines beginning with an exclamation mark), and comments at the end of lines to document the file.

8.5 Automatic Setting of Terminal Characteristics: TERDFL.SYS

The file TERDFL.SYS establishes default terminal characteristics for non-standard keyboards. The file is used whenever the user executes the SET TERMINAL/INQUIRE command and the system was unable to determine the terminal type. The system looks up the terminal number in the file and, if found, sets the characteristics that are on that line. If the terminal number is not found, the system displays an error message.

For most DIGITAL terminals, SET TERMINAL/INQUIRE can determine the type. The only supported exception is the LA36. The same may apply to older, unsupported terminals or terminals made by other manufacturers.

If you want the SET TERMINAL/INQUIRE command to work for terminals whose type cannot be determined, enter them in the TERDFL.SYS file. (It is not necessary to put any other supported terminals in the file.) If SET TERMINAL/INQUIRE can determine the type, it ignores any entry in TERDFL.SYS.

An example of a TERDFL.SYS file follows:

```
!
!      T E R D F L . S Y S
!
KBH10:-KBH15:/DEV=LA36
```

The format for each line of the file is the terminal device designator, a colon, and then the command qualifiers that define the characteristics of the keyboard. A range of terminals can be specified on one line as the low-end terminal device designator, an optional colon, a dash, the high-end terminal device designator, a mandatory colon, and then the qualifiers.

You can use blank lines, comment lines (lines beginning with an exclamation mark), and comments at the end of lines to document the file.

8.6 Setting Permanent Terminal Characteristics

The /PERMANENT qualifier lets you set the permanent characteristics of a terminal. For example, to set the characteristics of the dial-up line on keyboard 14 for the current time-sharing session, you log in to an account that has HWCFG privilege and type the command:

```
$ SET TERMINAL/PERMANENT KB14:/DEV=LA36
```

When you include the /PERMANENT switch with the SET TERMINAL command, the command takes effect immediately. If you do not have HWCFG privilege or the characteristics of the line conflict with the command, RSTS/E prints an error message and reprints the DCL prompt. The characteristics set by executing a SET TERMINAL/PERMANENT command are in effect until the next system shut down or until you change them. If you set characteristics without /PERMANENT, they revert to the standard characteristics on logout.

Use the /PERMANENT qualifier with all SET TERMINAL commands in START.COM.

8.7 Using the /NOBROADCAST and /BROADCAST Command Qualifiers

The /NOBROADCAST qualifier prevents your terminal or any terminal on the system from receiving broadcasts sent by the BROADCAST command. Use this when:

- You need to print a document on a letter-quality printer and do not want messages corrupting the printout
- You are using a text editor in screen mode at a dial-up terminal set at a low baud rate. Repainting the screen each time someone sends a message can be time consuming, especially at 300 baud.

Use the /BROADCAST qualifier to allow your terminal to receive messages again. Use the /NOBROADCAST qualifier to keep messages from printing on your own terminal. For example:

```
$ SET TERMINAL/NOBROADCAST
$ SHOW TERMINAL/BRIEF
Terminal: KB29:        Device_Type: VT52

Break            No Broadcast     Control=(R,T,C)  No Control=X
CRFill=0         No Delimiter     No Eight_bit     No Form
No Hostsync      Interactive      Lowercase        Resume=Control_C
Scope            Speed=9600       Tab              TTSync
Width=80
$
```

The SHOW TERMINAL command lets you verify that the command worked and that the rest of the terminal settings are set properly. If you decide to reverse this condition, use the /BROADCAST qualifier. For example:

```
$ SET TERMINAL/BROADCAST
$ SHOW TERMINAL/BRIEF
Terminal: KB29:        Device_Type: VT52

Break            Broadcast        Control=(R,T,C)  No Control=X
CRFill=0         No Delimiter     No Eight_bit     No Form
No Hostsync      Interactive      Lowercase        Resume=Control_C
Scope            Speed=9600       Tab              TTSync
Width=80
$
```

Your terminal can now receive BROADCAST messages.

You can also use the /NOBROADCAST and /BROADCAST qualifiers on other terminals on your system. Use the SET TERMINAL command as follows:

```
$ SET TERMINAL KB25:/NOBROADCAST/PERM
$ SHOW TERMINAL/BRIEF KB25:
Terminal: KB25:        Device_Type: VT100

Break            No Broadcast     Control=(R,T,C)  No Control=X
CRFill=0         No Delimiter     No Eight_bit     No Form
Hostsync         Interactive      Lowercase        Resume=Control_C
Scope            Speed=9600       Tab              TTSync
Width=80
$
```

8.8 Terminal Servers (LAT)

RSTS/E includes support that lets systems with DECnet/E connect terminals directly over the Ethernet from terminal servers. These terminal connections are called Local Area Transport (LAT). LAT lets users access your system directly from terminals connected to the following terminal servers:

- DIGITAL Ethernet Terminal Server (DECSA)
- DECserver 100
- DECserver 200
- DECserver 500
- DECserver 550
- VAXmate systems
- IBM-PC systems running DECnet-DOS

RSTS/E cannot access printers through terminal servers.

The DECSA and DECserver terminal servers require that software be loaded into them from a host system on the Ethernet. Since RSTS/E cannot load this software, it must be loaded from another system. The systems currently available to load this software are:

- VAX/VMS
- ULTRIX-32
- ULTRIX-32m
- DECsystem-10
- DECSYSTEM-20
- RSX-11M-PLUS
- Micro/RSX

8.8.1 Enabling LAT

To support LAT terminal servers, you must have DECnet/E and the Ethernet hardware. If you have these, RSTS/E automatically makes LAT available.

LAT support requires about 2K words of memory. If you are not using LAT on your Ethernet, use the following command to prevent LAT support in your system:

```
$ SET SYSTEM/NOLAT
```

After entering this command, you must shut the system down and restart it to change the state of LAT. To restore LAT support, use the SET SYSTEM/LAT command, followed by another shutdown and restart.

Use the SHOW SYSTEM command to determine the current state of LAT and the state that will take effect after the next reboot (if different from the current state).

Use the SHOW TERMINAL command to list the LAT server and port name for those connections coming from LAT terminal servers.

Use the following DCL commands to set parameters for the LAT software:

Command	Description
CREATE/SERVICE/LAT	Creates a LAT service
DELETE/SERVICE/LAT	Deletes a LAT service
SET NODE/LAT	Sets LAT node characteristics
SET SERVICE/LAT	Sets LAT service characteristics
SHOW COUNTERS/LAT	Shows LAT related counters
SHOW NODE/LAT	Shows LAT node characteristics
SHOW SERVICE/LAT	Shows LAT service characteristics
SHOW SESSIONS	Shows information about LAT sessions
SHOW TERMINAL_SERVERS/LAT	Shows terminal servers known to LAT
START/LAT	Starts LAT on an Ethernet device
STOP/LAT	Stops LAT on an Ethernet device

Each of these commands is described more fully at the end of this chapter.

8.8.2 Isolation of LAT Problems

To isolate LAT problems, use the DCL commands SHOW COUNTERS/LAT and SHOW COUNTERS/LAT/DEVICE, and the DECnet/E Network Control Program (NCP) utility. To invoke NCP, type:

```
$ RUN DECNET$:NCP
```

You should then get the NCP> prompt. The following NCP commands are particularly useful in isolating LAT problems:

```
NCP> SHOW LINE dev COUNTERS
```

```
NCP> LOOP CIRCUIT dev PHYSICAL ADDRESS ethernet-address
```

Note that *dev* is the name of the device being used (UNA-0 or QNA-0) and *ethernet-address* is the Ethernet address of the terminal server under question. See the *DECnet/E System Manager's Guide* for more information on these commands.

For more information on LAT activity, consult the console terminal. The LOGIN and LOGOUT commands automatically send the server and port names for LAT terminals to the OPSER program, which relays the names to the console. (If OPSER is not running, LOGIN and LOGOUT send the names to KB0:.)

The LOGIN command also sends the server and port names to [0,1]LOGIN.COM, which you can modify to respond to the information as you see fit. LOGIN.COM automatically passes the names to the group and user LOGIN.COM files in Parameter P5, and executes a SET TERMINAL/INQUIRE command for the LAT terminal.

8.9 Pseudo Keyboards

The monitor dynamically creates pseudo keyboards as they are needed and eliminates them when they are closed. Because of this, you have no chance to preset any of the terminal characteristics for these pseudo keyboards. The only time pseudo keyboards are not available is when all 128 keyboards are in use or when the system is very low on buffer space.

Some older applications require static pseudo keyboards instead of dynamic pseudo keyboards. OPSER BATCH requires static keyboards. To create these static pseudo keyboards, use the following command:

```
$ SET SYSTEM/PSEUDO_KEYBOARD=n
```

where *n* is the number of static pseudo keyboards needed. You can preset terminal characteristics for these pseudo keyboards with the SET TERMINAL/PERM command.

For these commands to have any effect, you must shut the system down and restart.

Use the SHOW SYSTEM command to find out the number of pseudo keyboards available the next time the system is rebooted.

8.10 Enabling Forms Management Support (FMS-11)

To enable or disable FMS, use the command:

```
$ SET SYSTEM/[NO]FMS
```

After you issue the SET SYSTEM command, you must shut the system down and restart before FMS is available (or becomes unavailable).

Use the SHOW SYSTEM command to find out the current state of FMS and the state that will occur the next time the system is rebooted.

8.11 DCL Command Descriptions

The following sections explain the commands available for managing terminals: SET TERMINAL, SHOW TERMINAL, and the LAT-related commands.

8.11.1 CREATE/SERVICE/LAT Command

Creates a service which is then offered by the LAT host node. You need the SWCFG privilege to use the CREATE/SERVICE/LAT command. Example:

```
$ CREATE/SERVICE/LAT EDERE
```

Format

CREATE/SERVICE/LAT *service_name*

Command Qualifiers	Defaults
/IDENTIFICATION="service_id"	none
/[NO]LOG	/NOLOG
/STATIC_RATING=rating	/STATIC_RATING=255

Prompts

Service name:

Command Parameters

service_name

Gives a name for the service being created. Service names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). Typically, the service name is the same as the DECnet/E node name.

Command Qualifiers

/IDENTIFICATION="service_id"

Specifies a description for the service. This description is used to further identify the service being offered. It may be up to 64 characters in length.

/[NO]LOG

Specifies whether or not a confirmation message is displayed indicating that the requested action was taken. The default is /NOLOG.

/STATIC_RATING=rating

Specifies the static rating associated with the service. The valid range is 0 to 255. The default is /STATIC_RATING=255.

8.11.2 DELETE/SERVICE/LAT Command

Deletes a service currently offered by the LAT host node. You need the SWCFG privilege to use the DELETE/SERVICE/LAT command. For example:

```
$ DELETE/SERVICE/LAT EDERE
```

Format

DELETE/SERVICE/LAT *service_name*

Command Qualifiers	**Defaults**
/[NO]LOG	/NOLOG

Prompts
Service name:

Command Parameters

service_name
Specifies the service being deleted. Service names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($).

Command Qualifiers

/[NO]LOG
Specifies whether or not a confirmation message is displayed indicating that the requested action was taken. The default is /NOLOG.

8.11.3 SET NODE/LAT Command

Sets or changes the initial LAT host node characteristics. You must have the SWCFG privilege to use the SET NODE/LAT command. Example:

```
$ SET NODE/LAT/IDENTIFICATION="Welcome to LAT on EDERE"
```

Format

SET NODE/LAT

Command Qualifiers	Defaults
/DISABLE={group_list | ALL}	none
/ENABLE={group_list | ALL}	none
/IDENTIFICATION="node_id"	none
/[NO]LOG	/NOLOG
/MULTICAST_TIMER=seconds	/MULTICAST_TIMER=60

Prompts

none

Command Qualifiers

/DISABLE=group_list

Removes the specified group codes from the list of group codes associated with the LAT host node. The group_list can include multiple codes, separated by commas. For example, /DISABLE=(0,1,4,10) disables group codes 0,1,4, and 10. Use the keyword ALL to indicate codes 0 to 255. Specify the /DISABLE qualifier no more than once on a command line. No error is returned if a specified group code is not enabled.

/ENABLE=group_list

Adds the specified group codes to the list of group codes associated with the LAT host node. The group_list can include multiple codes, separated by commas. For example, /ENABLE=(5,6,8,10) enables group codes 5,6,8, and 10. Use the keyword ALL for codes 0-255. Specify the /ENABLE qualifier no more than once on a command line. No error is returned if a specified group code is already enabled.

/IDENTIFICATION="node_id"

Specifies a description for the LAT host node. You can use this description as an announcement string or to further identify the node. It may be up to 64 characters.

/[NO]LOG

Specifies whether or not a confirmation message is displayed indicating that the requested action was taken. The default is /NOLOG.

/MULTICAST_TIMER=seconds

Specifies the interval, in seconds, between the multicast messages sent for service announcements. The minimum value is 10 seconds, the maximum value is 255 seconds. The default is 60 seconds.

8.11.4 SET SERVICE/LAT Command

Modifies the information associated with a particular service being offered by th node. You need the SWCFG privilege to use the SET SERVICE/LAT command. Example:

```
$ SET SERVICE/LAT/STATIC_RATING=40 EDERE
```

Format

SET SERVICE/LAT *service_name*

Command Qualifiers	Defaults
/IDENTIFICATION="service_id"	none
/[NO]LOG	/NOLOG
/STATIC_RATING=rating	none

Prompts

Service name:

Command Parameters

service_name

Identifies the service being modified. Service names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). Typically, th service name is the same as the node name.

Command Qualifiers

/IDENTIFICATION="service_id"

Modifies the description for the service. You can use this description to further identify the service being offered.

/[NO]LOG

Specifies whether or not a confirmation message is displayed indicating the requested action was taken. The default is /NOLOG.

/STATIC_RATING=rating

Modifies the static rating associated with the service.

8.11.5 SET SYSTEM Command

The SET SYSTEM command sets the default system characteristics. You need one or more privileges to use this command (see the discussion for each command qualifier).

See Chapter 14 for additional SET SYSTEM command qualifiers.

Format

SET SYSTEM

Command Qualifiers	Defaults
/[NO]FMS	none
/[NO]LAT	none
/PSEUDO_KEYBOARDS=n	/PSEUDO_KEYBOARDS=4

Command Qualifiers

/[NO]FMS
Enables FMS the next time the system restarts. The /NOFMS qualifier disables FMS. You need the SWCFG privilege to use this qualifier.

/[NO]LAT
Enables LAT support the next time the system restarts. The /NOLAT qualifier disables LAT support, saving 2K words of memory. You need the SWCFG privilege to use this qualifier.

/PSEUDO_KEYBOARDS=n
Specifies the maximum number, n, of static psuedo keyboards available on the system the next time the system restarts. n can range from 1 to 127. You need the SWCFG privilege to use this qualifier.

8.11.6 SET TERMINAL Command

Sets characteristics for terminals attached locally to a RSTS/E system or for terminals connected by dialup lines. Users can use the SET TERMINAL command to set characteristics for their own terminals. You need HWCFG privilege to define characteristics for other terminals. Example:

```
$ SET TERMINAL/ADVANCED_VIDEO KB04
```

Format

SET TERMINAL *[KB[c]nn[:]]*

Command Qualifiers	Defaults
/[NO]132_COLUMNS	none
/[NO]ADVANCED_VIDEO	none
/[NO]ALT_MODE	none
/[NO]ANSI	none
/[NO]ANSWERBACK	none
/[NO]AUTOBAUD	none
/[NO]BREAK	none
/[NO]BROADCAST	none
/BUFFER_QUOTA=n	/BUFFER_QUOTA=180
/[NO]CONTROL=[(option[,option,...])]	none
/[NO]CRFILL[=n]	none
/[NO]DELIMITER[=c]	none
/[DEVICE_TYPE=]terminal type	none
/[NO]DIALUP	none
/[NO]EIGHT_BIT	none
/[NO]ESCAPE_SEQUENCE	none
/[NO]FORM_FEED	none
/HARDCOPY	none
/[NO]HOST_SYNC	none
/INQUIRE	none
/[NO]INTERACTIVE	none
/[NO]KATAKANA	none
/[NO]LOADABLE_CHARACTERS	none
/[NO]LOCAL_ECHO	none
/LOWERCASE[=INPUT | OUTPUT]	none
/[NO]PARITY[=option]	none
/PERMANENT	none
/[NO]PRINTER_PORT	none
/[NO]REGIS	none
/RESET	See discussion
/[NO]RESTRICT	none
/RESUME=[ANY | CONTROL_C]	none
/SCOPE	none
/SELECT_ERASE	none
/SETUP=filename	none
/[NO]SIXEL	none
/SPEED=(input[,output])	none
/[NO]TAB	none

/[NO]TTSYNC	none
/TYPE=n	none
/[NO]UP_ARROW	none
/UPPERCASE[=INPUT | OUTPUT]	none
/[NO]USER_DEFINED_KEYS	none
/WIDTH=n	none

Prompts

none

Command Parameters

[KB[c]nn[:]]
Indicates the terminal to set the characteristics for. If you do not specify a terminal, RSTS/E sets the the characteristics of the terminal that issued the command. You must have HWCFG privilege to set the characteristics for another user's terminal.

Command Qualifiers

/[NO]132_COLUMNS
If you specify /132_COLUMNS, the terminal displays single-width characters in an array of 24 lines by 132 columns. If you specify /NO132_COLUMNS, the terminal displays single-width characters in an array of 24 lines by 80 columns.

/[NO]ADVANCED_VIDEO
If you specify /ADVANCED_VIDEO, it means the terminal has the advanced video option (AVO). The advanced features include character attributes (blinking, bolding, ..., and so on), screen memory, and ROM sockets. If you specify /NOADVANCED_VIDEO, it means the terminal does not support AVO.

/[NO]ALT_MODE
If you specify /ALT_MODE, the system treats ASCII 027, 125, and 126 as ESCAPE. If you specify /NOALT_MODE, the system treats only ASCII 027 code as ESCAPE. See /LOWERCASE for more information on the use of this qualifier.

/[NO]ANSI
If you specify /ANSI, it means the terminal supports ANSI escape sequences. If you specify /NOANSI, it means the terminal does not support ANSI escape sequences.

/[NO]ANSWERBACK
If you specify /ANSWERBACK, the terminal sends verifying identification to electronic messaging services such as Telex or TWX. Users on dialup lines must run the LOGIN program by pressing Return. /ANSWERBACK cannot be used with /AUTOBAUD. If you specify /NOANSWERBACK, the terminal does not send the identification message. The terminal can be used for normal sessions on the system in either case.

/[NO]AUTOBAUD
If you specify /AUTOBAUD, the system monitors the terminal line to detect and set baud rates without requiring the user to manually set or determine the terminal's speed. If you specify /NOAUTOBAUD, the system does not set the terminal line's speed without manual intervention.

/[NO]BREAK

If you specify /BREAK, the system translates the BREAK key as a Ctrl/C character (ASCII 3). If you specify /NOBREAK, the system translates the BREAK key as a null character (ASCII 0).

/[NO]BROADCAST

The /BROADCAST qualifier enables the system broadcast capability for the terminal. That is, RSTS/E displays messages from the SEND system function call on the terminal. The /NOBROADCAST qualifier disables the system broadcast capability for the terminal.

/BUFFER_QUOTA=n

Indicates the number of input characters that you can buffer before the system issues an XOFF character to your terminal to stop transmission. The buffer quota value is a multiple of 30. The allowable values range from 180 to 7650. By default, the system uses a quota of 180 characters. The system uses one small buffer for each 30 characters. Thus, the default buffer quota of 180 characters uses a maximum of six small buffers for terminal input. Changing a terminal's input buffer quota is useful for high-speed input terminals.. However, if there are not enough small buffers available, a terminal may not be able to reach its full quota. You need HWCFG privilege to use the /BUFFER_QUOTA qualifier.

/[NO]CONTROL=[(option[,option,...])]

If you specify /CONTROL=C, the system traps a Ctrl/C typed by the user and handles it accordingly. If you specify /NOCONTROL=C, the system ignores a Ctrl/C typed by the user.

If you specify /CONTROL=R, the system enables the Ctrl/R retype facility. If you specify /NOCONTROL=R, the system disables the Ctrl/R retype facility.

If you specify /CONTROL=T, the system enables the Ctrl/T job status display facility. If you specify /NOCONTROL=T, the system disables the Ctrl/T job status display facility.

If you specify /CONTROL=X, the system traps a Ctrl/X typed by the user and clears the user's entire type-ahead buffer. If you specify /NOCONTROL=X, the system interprets a Ctrl/X typed by the user as data.

If you want to set or reset more than one option on a single command line, you can combine them in a list. For example: /CONTROL=(R,T,C) or /NOCONTROL=(C,X).

/[NO]CRFILL[=n]

If you specify /CRFILL, the system sets the carriage return fill factor to n, where n is between 0 and 6. If the argument is not specified or is zero, or if /NOCRFILL is specified, the system does not generate any fill characters.

/[NO]DELIMITER[=c]

If the argument is a character within quotation marks (for example /DELIMITER="\ "), the character within quotation marks becomes the private delimiter.

If the argument is a number (for example /DELIMITER=9), the private delimiter is the ASCII character whose decimal value is the number. This allows for any ASCII character whose decimal value is between 1 and 127 to be a private delimiter.

If you specify /NODELIMITER, the system removes the private delimiter that was previously set.

/[DEVICE_TYPE=]terminal type

Sets the default characteristics for the terminal type specified or, if it is a name of a user-defined macro, sets the characteristics defined in the macro. This command recognizes the following terminal types:

```
LA12 LA50 LA180S VT100 VT125 VT240
LA34 LA75 VK100 VT101 VT131 VT320
LA36 LA100 VT52 VT102 VT132 VT330
LA38 LA120 VT55 VT105 VT220 VT340
```

See the section "User-Defined Terminal Characteristics: TERMAC.SYS" for a description and example of the user-defined macro file.

/[NO]DIALUP

Indicates if dialup control is enabled for the specified terminal.

/[NO]EIGHT_BIT

If you specify /EIGHT_BIT, it means the terminal supports eight bit characters. If you specify /NOEIGHT_BIT, it means the terminal does not support eight bit characters.

/[NO]ESCAPE_SEQUENCE

If you specify /ESCAPE_SEQUENCE, the system treats an ESC character (value 27) as an indication of an incoming escape sequence. The system does not echo the character and or any of the characters in the sequence. If you specify /NOESCAPE_SEQUENCE, the system treats an ESC character (value 27) as a line terminator and echoes it as a $ character.

/[NO]FORM_FEED

The /FORM_FEED qualifier indicates that the hardware has form feed and vertical tab capabilities. The system transmits form feed and vertical tab characters without translation. The /NOFORM_FEED qualifier indicates that the hardware does not have form feed and vertical tab capabilities. The system transmits four line feed characters in place of a form feed or vertical tab character.

/HARDCOPY

If you specify /HARDCOPY, it means the terminal is not a cathode ray terminal (CRT) display device. The system echoes a DEL character (RUBOUT) by printing a backslash (\) and the last character typed and removes the last character typed from the terminal input buffer. Later DEL characters cause the next to last characters to be sequentially printed and removed from the terminal input buffer until a character other than DEL is received. As a result, the system echoes another \ to delimit the erased characters and then echoes the correct character.

If you specify /HARDCOPY and /[NO]TTSYNC is not specified, /NOTTSYNC is set by default.

/[NO]HOSTSYNC

Specify /HOSTSYNC if the terminal has special hardware that lets the computer interrupt transmission of characters from the terminal by sending the terminal an XOFF character (value 19). Similarly, the computer instructs the terminal to resume transmission of characters by sending the terminal an XON character (value 17). The terminal hardware must respond to XOFF and XON characters by stopping and resuming transmission.

Specify /NOHOSTSYNC if the terminal does not have the hardware required for the HOSTSYNC feature.

/INQUIRE

Queries the terminal by sending an ANSI ESCAPE identifying sequence, then determines the terminal type and sets the appropriate terminal characteristics. If the terminal does not respond within five seconds, the system searches the file TERDFL.SYS to see if there are default characteristics for this terminal. If an entry is found, the system sets the specified characteristics. If not, the system displays an error message and the terminal characteristics are unchanged.

NOTE

> This qualifier causes the terminal's type-ahead buffer to be cleared. LA36 terminals do not support this feature.

See the section "Automatic Setting of Terminal Characteristics: TERDFL.SYS" for more information on using /INQUIRE with terminals that do not respond to the identifying sequence.

/[NO]INTERACTIVE

Indicates that you can use the terminal to log in to the system. If you specify /NOINTERACTIVE, the system ignores any characters input from the terminal while it is not assigned to another job or not opened. Use this attribute for KB devices set up as printing devices.

/[NO]KATAKANA

Indicates that the terminal supports the Katakana character set.

/[NO]LOADABLE_CHARACTERS

Indicates that the terminal supports dynamically redefinable character sets (for example, changing fonts).

/[NO]LOCAL_ECHO

If you specify /LOCAL_ECHO, the system does not echo characters it receives from the terminal. This is only used for a terminal that does its own local echo.

If you specify /NOLOCAL_ECHO, characters generated are sent only to the computer. Therefore, the system echoes each character it receives so the characte is displayed locally and then translates certain characters to perform the proper action. For example, the system echoes a CR character as a carriage return and line feed sequence.

/LOWERCASE[=INPUT | OUTPUT]

The /LOWERCASE=INPUT qualifier indicates that the system enables lowercase input from the terminal. The /LOWERCASE=OUTPUT qualifier indicates that the system enables lowercase output to the terminal.

If you specify /LOWERCASE with no argument, the system enables lowercase input and output. If you specify /LOWERCASE or /LOWERCASE=INPUT and /[NO]ALTMODE is not specified, the system sets /NOALTMODE by default.

/[NO]PARITY=EVEN | ODD

If you specify /NOPARITY, the system ignores the parity bit on characters it receives and treats the parity bit on characters it transmits to the terminal as if the parity bit were a data bit.

If you specify /PARITY=EVEN, the system sends characters to the terminal with the parity bit properly set for even parity but ignores the parity bit on characters it receives.

If you specify /PARITY=ODD, the system sends characters to the terminal with the parity bit properly set for odd parity but ignores the parity bit on characters it receives.

/PERMANENT

If you specify /PERMANENT, the other qualifiers on the same command line become the terminal's permanent characteristics. Permanent characteristics are the characteristics of the terminal when it is not in use. The characteristics of the terminal revert to the permanent characteristics when the user logs out.

You can use /PERMANENT to specify any characteristics of any terminal, including dialup controlled terminals. You must use /PERMANENT with /ANSWERBACK, /AUTOBAUD, /DIALUP, and /INTERACTIVE. You must have HWCFG privilege to specify this qualifier.

/[NO]PRINTER_PORT

Indicates that the terminal has a printer port.

/[NO]REGIS

Indicates that the terminal supports the Remote Graphic Instruction Set (ReGIS).

/RESET

Resets the terminal's characteristics to the permanent characteristics.

/[NO]RESTRICT

If you specify /RESTRICT, it means that only users with DEVICE privilege can access the specified terminal. If you specify /NORESTRICT, it means that any users can access the specified terminal.

This qualifier only reflects OPENs and ASSIGNs of the device: not LOGINs.

/RESUME=[ANY | CONTROL_C]

Defines the XON/XOFF processing. The argument ANY enables type out and echo when any character is typed after XOFF. ANY is valid only on LA36 and VT52 terminals. The argument CONTROL_C enables type out and echo only when you type XON or Ctrl/C after XOFF.

/SCOPE

Specify /SCOPE if the terminal is a CRT display terminal that has the following characteristics:

- Conforms to synchronization as described under the /TTSYNC command.
- System echoes a DEL character (RUBOUT) as backspace, space, and backspace sequence.

If you specify /SCOPE, and /[NO]TTSYNC is not specified, /TTSYNC is set by default.

/[NO]SELECT_ERASE

Indicates if the terminal supports the selectively erasable character attribute.

If the terminal supports this feature, the terminal recognizes two forms of the erase in line and erase in display commands. In one form all characters within the set parameter are erased regardless of logical attributes. The second form changes all characters designated with the selectively erasable character attribute into spaces.

/SETUP=filespec

Sends the specified file's data to the terminal, in binary mode. If you specify a file name only, the system assumes the file is in the user's account on the system disk with a file type of .ESC. The user must have access to the file or an error results. Use this command to initialize a terminal for which you can make special software settings.

/[NO]SIXEL

Indicates if the terminal supports Sixel Graphics. Sixel Graphics transfer binary graphic images between the host and the terminal or the terminal and a printer.

/SPEED=(input[,output)]

Specifies the speed at which the terminal sends and receives data.

You can specify the speed as a single value to set the input and output baud rates to the same speed. To specify a split speed for input and output, specify /SPEED=(n,m), where the values n and m indicate the input (terminal to host) and output (host to terminal) baud rates, respectively.

NOTE

> The /SPEED qualifier works on CXA, CXB, CXF, DH, DHF, DHQ, DHV, DHU, DZ, DZQ, and DZV terminals. Split speed does not work on DZ, DZQ, and DZV terminals.

You can use the keywords EXTA and EXTB to specify a baud rate associated with the DH11 External Clock A and External Clock B baud rates, respectively.

See the section "Terminal Line Speed Characteristics File: TERSPD.SYS" for a description and example of the speed file.

/[NO]TAB

/TAB indicates the terminal hardware has tab capability. The system transmits TAB characters without translation.

/NOTAB indicates the terminal hardware does not have tab capability. To move to the next tab stop, the system transmits the correct number of space characters instead of transmitting a TAB character.

/[NO]TTSYNC

If you specify /TTSYNC, the terminal obeys the following synchronization standard: if the terminal sends an XOFF character (equivalent to the Ctrl/S combination), the computer interrupts transmission until the terminal sends either an XON character (equivalent to the Ctrl/Q combination) or a Ctrl/C combination.

If you specify /NOTTSYNC, XON and XOFF characters sent by the terminal have no special meaning.

/TYPE=n

Specifies the value of the terminal's type code. This type code coincides with a particular terminal type and the SHOW TERMINAL command uses it to display the terminal type. This qualifier is best used in the TERMAC.SYS file to associate a terminal with a user-defined macro.

Allowable values are 0 to 255. The values 0 to 128 are reserved for DIGITAL use.

/[NO]UP_ARROW

If you specify /UP_ARROW, the system echoes a control and graphic character combination as the ^ character (value 94) followed by the proper graphic character. For example, Ctrl/E prints out as ^E.

If you specify /NOUP_ARROW, the system echoes the control and graphic character combination as is.

/UPPERCASE[=INPUT | OUTPUT]

The /UPPERCASE=INPUT qualifier indicates that the system disables lowercase input from the terminal. The /UPPERCASE=OUTPUT qualifier indicates that the system disables lowercase output to the terminal.

If you specify /UPPERCASE with no argument, the system disables lowercase input and output. If you specify /UPPERCASE or /UPPERCASE=INPUT and /[NO]ALTMODE is not specified, the system sets /ALTMODE by default.

/[NO]USER_DEFINED_KEYS

Indicates if the terminal supports User Defined Keys (UDKs). User Defined Keys let you save a full command string and mvoke it with a single key.

/WIDTH=n

Sets the width of the print line for the terminal to n. The width can be between 1 and 254. As a result, the system automatically generates a carriage return/line feed combination if n printing characters have been printed or echoed without a carriage return/line feed combination and another printing character is to be transmitted.

If the terminal executing the program is in the VT100/200 family and the print line width is changed, then the appropriate ESC sequences are sent to the terminal to change the width of the screen accordingly. For example, if the width changes from 80 or less to 81 or more, then the width of the screen is set to 132; if the width changes from 81 or more to 80 or less, the width of the screen is set to 80.

8.11.7 SHOW COUNTERS/LAT Command

Displays information about counters relevant to managing LAT on the host node. You need the SWCTL privilege to use the SHOW COUNTERS/LAT command. There are three types of counter information: LAT host node counters, terminal server counters, and Ethernet portal counters. If you do not specify one of these, the system displays information on node counters as the default. For example:

```
$ SHOW COUNTERS/LAT

     LAT Host Node Counters
              Seconds since last zeroed        : 882294
              Messages Transmitted             : 614796
              Bytes Transmitted                : 13103390
              Messages Received                : 579669
              Bytes Received                   : 26923552
              Messages Retransmitted           : 847
              Messages Received Out of Sequence : 0
              Invalid Messages Received        : 8
              Invalid Slots Received           : 0
              Illegal Messages Received        : 0
              Illegal Slots Received           : 0
              Resource Allocation Errors       : 0
              Transmit Errors                  : 0
              Circuit Timeouts                 : 1
```

Format

SHOW COUNTERS/LAT *[server_name]*

Command Qualifiers	Defaults
/DEVICE	none
/OUTPUT=output_file	none
/NODE	none
/TERMINAL_SERVERS	none
/ZERO	none

Prompts

none

Command Parameters

[server_name]

Specifies the terminal server for which counter information displays. Server names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). If you want to examine a terminal server, use the /TERMINAL_SERVERS qualifier. If you do not use that qualifier or use the /NODE qualifier instead, the SHOW COUNTERS/LAT command displays information on the host node counters.

Command Qualifiers

/DEVICE

Displays the Ethernet device counters. These device counters pertain only to the portal which Ethernet has open for LAT service. The device parameter is optional. If you leave it out, you get information on counters for all Ethernet portals open for LAT service.

/OUTPUT=output_file

Sends the display of the requested counters to the output file specified.

/NODE

Displays the counters for the LAT host node. /NODE is the default if neither /DEVICE nor /TERMINAL_SERVERS is specified.

/TERMINAL_SERVERS

Displays the counters for the terminal server(s) known to the LAT host node. If a server name parameter is specified, only the counters for that server are displayed. If no parameter is specified, then counters for all servers known to the LAT host node are displayed.

/ZERO

Zeroes the counters after they are displayed. This qualifier acts only on the counter(s) that are displayed.

8.11.8 SHOW NODE/LAT Command

Displays information about the current LAT host node characteristics. The information displayed includes the LAT version number, the node name and identification, the multicast timer, and the groups to which the LAT host node is associated. For example:

```
$ SHOW NODE/LAT

Node Name:       EDERE
Identification:  RSTS V9.7-08 Software at its finest
Multicast Timer: 60
Group Codes:     141  175  210  211  216

LAT is enabled on line XE0:
```

Format

SHOW NODE/LAT

Command Qualifiers	Defaults
/OUTPUT=output_file	none

Prompts
none

Command Qualifiers

/OUTPUT=output_file
Requests that the output of this display be put into a file, and not be displayed at the terminal.

8.11.9 SHOW SERVICE/LAT Command

Displays information about the service(s) offered by the LAT host node. If you specify a service name, only that service displays. Otherwise, all services offered by the LAT host will be displays. For example:

```
$ SHOW SERVICE/LAT

Service:        EDERE
Identification: Unauthorized use prohibited
Static rating:  255
```

Format

SHOW SERVICES/LAT *[service_name]*

Command Qualifiers	Defaults
/OUTPUT=output_file	none

Prompts

none

Command Parameters

[service_name]
Specifies the service for which information is displayed. Service names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). Typically, the service name is the same as the node name. If you do not specify a service, the SHOW SERVICE/LAT command displays all the services offered by the LAT host.

Command Qualifiers

/OUTPUT=output_file
Requests that the output of this display be put into a file, and not be displayed at the terminal.

8.11.10 SHOW SESSIONS/LAT Command

Displays information about the sessions which are currently running LAT on the host node. The information includes the RSTS/E keyboard number that the user is logged in to, the terminal server that the user's terminal is physically connected to, and the name of the service the user is currently running. For example:

```
$ SHOW SESSIONS/LAT
 KB43    Port: TWA30             Server: BEANEY          Service: EDERE
 KB46    Port: WTA4              Server: JTSVAX          Service: EDERE
 KB48    Port: PORT_7            Server: M12M12          Service: EDERE
 KB50    Port: PORT_53           Server: M12M12          Service: EDERE
 KB47    Port: PORT_68           Server: M12M12          Service: EDERE
 KB44    Port: TWA62             Server: BRAIDE          Service: EDERE
 KB45    Port: TWA63             Server: BRAIDE          Service: EDERE
 KB54    Port: WTA6              Server: SHERMN          Service: EDERE
 KB55    Port: WTA9              Server: SHERMN          Service: EDERE
 KB49    Port: TWA87             Server: BOTTLE          Service: EDERE
 KB51    Port: PORT_88           Server: M12M11          Service: EDERE
 KB52    Port: PORT_3            Server: M12M26          Service: EDERE
```

Format

SHOW SESSIONS/LAT *[server_name]*

Command Qualifiers	Defaults
/DEVICE=Ethernet_device:	none
/LAT	none
/OUTPUT=output_file	none

Prompts

none

Command Parameters

[server_name]

Specifies the server for which information is displayed. Server names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). If you do not specify a server, the SHOW SESSIONS/LAT command displays all the servers operating on the LAT host.

Command Qualifiers

/DEVICE=device:

Restricts the search to those users who are currently running LAT on the particular Ethernet device specified.

/LAT

Displays LAT sessions rather than other sessions. /LAT is the default; SHOW SESSIONS gives you the same display as SHOW SESSIONS/LAT.

/OUTPUT=output_file

Sends the output of this display to a file instead of displaying it at the terminal.

8.11.11 SHOW TERMINAL Command

Displays the current characteristics of a specific terminal. You can change each of these characteristics with a corresponding option of the SET TERMINAL command. The following examples show the output format for the SHOW TERMINAL command:

```
$ SHOW TERMINAL/BRIEF
Terminal: KB26:          Device_Type: VT220

    No Break        Broadcast          Control=(R,T,C)     No Control=X
    CRFill=0        No Delimiter       Eight_bit           No Form
    Hostsync        Interactive        Lowercase           Resume=Control_C
    Scope           Speed=9600         Tab                 TTSync
    Width=80

$ SHOW TERMINAL/FULL
Terminal: KB26:          Device_Type: VT220

    132_Columns          Advanced_Video       No Alt_Mode        ANSI

    No Answerback        No Autobaud          No Break           Broadcast

    Buffer_Quota=180     Control=(R,T,C)      No Control=X       CRFill=0
    No Delimiter         No Dialup            Eight_bit          No Escape_Sequence
    No Form              Hostsync             Interactive        No Katakana
    Loadable_Characters  No Local_Echo        Lowercase          No Parity
    Printer_Port         No ReGis             Restricted         Resume=Control_C
    Scope                Select_Erase         No Sixel           Speed=9600
    Tab                  TTSync               Up_Arrow           User_Defined_Keys
    Width=80
$
```

Format

SHOW TERMINAL *[KB[c]nn[:]]*

Command Qualifiers	Defaults
/ALL	Current Terminal
/BRIEF	/BRIEF
/FULL	/BRIEF
/[NO]PERMANENT	/NOPERMANENT

Prompts

none

Command Parameters

[KB[c]nn[:]]

Indicates the terminal described in the display. When you do not specify a terminal, the system displays the characteristics of the terminal executing the command. You must have HWCFG privilege to specify another user's terminal.

Command Qualifiers

/ALL

Displays the characteristics of all the terminals configured on the system. If a terminal is specified along with /ALL, the terminal specification overrides /ALL.

/BRIEF

If you specify /BRIEF, or neither /BRIEF nor /FULL, then a subset of the full display is shown.

/FULL

If you specify /FULL, the full display is shown.

/[NO]PERMANENT

If you specify /PERMANENT, the system displays the permanent characteristics of the terminal. If you specify /NOPERMANENT, the system displays the current characteristics of the terminal.

8.11.12 SHOW TERMINAL_SERVERS/LAT Command

Displays information about the terminal server(s) known to the LAT host node. The server name is optional; if you omit it, information displays about all servers known to the LAT. For example:

```
$ SHOW TERMINAL_SERVERS/LAT BEANEY
 Server:  BEANEY
   Active sessions:        1
   Ethernet address:       AA-00-00-40-D0-0A
   Keep alive timer:       20 seconds
   Maximum buffer size:    1518 bytes
   LAT Protocol:           V5.0
```

Format

SHOW TERMINAL_SERVERS/LAT *[server_name]*

Command Qualifiers	Defaults
/OUTPUT=output_file	none

Prompts

none

Command Parameters

[server_name]

Specifies the server for which information is displayed. Server names may have as many as 16 characters, including the multinational alphanumeric characters and the characters period (.), underscore (_), hyphen (-), and dollar sign ($). If you do not specify a server, the SHOW TERMINAL_SERVERS/LAT command displays all the servers operating on the LAT host.

Command Qualifiers

/OUTPUT=output_file

Requests that the output of this display be put into a file, and not be displayed at the terminal.

8.11.13 START/LAT Command

The START/LAT command enables LAT service. You can start LAT on a particular Ethernet device by specifying the device on the command line. If you specify no device, LAT starts on all the Ethernet devices found. You must issue a SET NODE/LAT command before issuing the first START/LAT command. You must have the SWCTL privilege to use the START/LAT command. For example:

```
$ START/LAT XE0:
```

Format

START/LAT *Ethernet_device:*

Command Qualifiers	**Defaults**
/[NO]LOG	/LOG

Prompts

Ethernet device:

Command Parameters

Ethernet_device:

Specifies the Ethernet portal to use for LAT. The device name is usually XE0: or XH0:.

Command Qualifiers

/[NO]LOG

Specifies whether or not a confirmation message is displayed indicating whether or not LAT has started up. The default is /LOG.

8.11.14 STOP/LAT Command

The STOP/LAT command disables LAT service on a particular Ethernet device or on all Ethernet devices. If no Ethernet device parameter is specified, LAT is disabled on all Ethernet devices. Jobs detach if they are currently running on Ethernet devices that get disabled. You need the SWCTL privilege to use the STOP/LAT command. For example:

```
$ STOP/LAT XE0:
```

Format

STOP/LAT *[Ethernet_device:]*

Command Qualifiers	Defaults
/[NO]LOG	/LOG

Prompts

none

Command Parameters

[Ethernet_device:]
Specifies the Ethernet portal on which LAT is disabled. If you do not specify a portal, the STOP/LAT command disables LAT on all portals. Typically, there is only one LAT portal, XE0: or XH0:.

Command Qualifiers

/[NO]LOG
Specifies whether or not a confirmation message is displayed indicating whether or not LAT has stopped. The default is /LOG.

Chapter 9

Managing Print/Batch Services

This chapter gives background information on the Print/Batch Services (PBS) package. PBS contains routines to help you manage your RSTS/E V9.0 and later print and batch servers.

The chapter then describes the DCL commands for using PBS. Table 9–1 summarizes the DCL commands.

Table 9–1: DCL Commands for the Print/Batch Services Package

Command	Description
	Package-Related
START/QUEUE/MANAGER	Starts the Print/Batch Services package
STOP/QUEUE/MANAGER	Shuts down the Print/Batch Services package
	Queue-Related
ASSIGN/QUEUE	Assigns a queue to a server
CLOSE/QUEUE	Closes a queue, preventing it from accepting additiona entries
DEASSIGN/QUEUE	Deassigns a queue from a server, or all servers
DELETE/QUEUE	Deletes a queue
INITIALIZE/QUEUE	Defines a print or batch queue
OPEN/QUEUE	Opens a queue, allowing entries to be placed on it
SET QUEUE	Modifies the attributes of one or more queues
SHOW QUEUE	Displays the attributes and status of one or more queues
START/QUEUE	Starts a queue, allowing entries to be routed to server for processing
STOP/QUEUE	Stops a queue, preventing entries from being routed t servers for processing
	Server-Related
DELETE/SERVER	Deletes a print or batch server
INITIALIZE/SERVER	Defines a print or batch server
SET SERVER	Modifies the attributes of one or more servers

(continued on next page

Table 9–1 (Cont.): DCL Commands for the Print/Batch Services Package

	Server-Related
SHOW SERVER	Displays the attributes and status of one or more servers
START/SERVER	Starts a print or batch server that was stopped
STOP/SERVER	Stops a print or batch server immediately or when a specified event occurs
	Entry-Related
DELETE/ENTRY	Deletes one or more print or batch entries selected by entry-spec or entry-number
PRINT	Queues a request for printing one or more files
SET ENTRY	Modifies the attributes of one or more print or batch entries
SHOW ENTRY	Displays the attributes and status of one or more print or batch entries
SUBMIT	Queues a request for batch processing one or more command files

9.1 Managing the Print/Batch Services Package

The Print/Batch Services (PBS) package provides several important services for RSTS/E users and system managers. The major components of PBS are:

- Queue Manager
- Print Servers
- Batch Servers
- User Interface

The Queue Manager accepts all user requests, controls all print and batch queues, schedules jobs to run, and creates new jobs as needed to process requests.

Print servers process print requests on selected printers. Both line printers (LP) and terminals (KB) can be handled by print servers.

Batch servers control execution of submitted command procedures, using the DCL command file processor (see *RSTS/E Guide to Writing Command Procedures*). You can establish multiple batch servers to process several command procedures at the same time.

The user interface is how an application program communicates with PBS. It lets the user program issue requests for print and batch services. User Request Packets (see *RSTS/E Programming Manual*) provide this link between PBS and the user application programs.

9.1.1 Queue Manager

The Queue Manager module is an integral part of PBS. The term queue refers to a list of pending print or batch requests. The term entry describes a request on a queue. Normally, PBS adds new entries to the end or tail of a queue. PBS processes an entry when it reaches the front or head of the queue.

PBS creates an entry on a queue in response to a user request for print or batch services. The entry remains on a queue until its requested job is completed, or until it is explicitly deleted by an operator or user. PBS assigns a priority to every queue entry which establishes its position on the queue relative to other entries. For example, PBS inserts a high-priority entry in the queue ahead of existing lower priority entries instead of at the end.

The system file PBS$:PBS.SYS contains all the queue definitions. You define a new queue with the INITIALIZE/QUEUE command. Once you define a queue it remains in the file until it is explicitly deleted. You delete a queue with the DELETE/QUEUE command.

9.1.1.1 Queue Types

PBS uses two types of queues:

- Print—The PRINT command creates entries in a print queue
- Batch—The SUBMIT command creates entries in a batch queue

You can define several queues for each type, but each queue can only have a single type associated with it. Note that once you define a queue as a print or batch queue, you cannot change its type.

9.1.1.2 Queue Names

Every queue must have a unique name. This name can consist of 1 to 9 characters from the set A-Z, 0-9, $, and _, and must begin with an alphanumeric character. PBS always forces lowercase characters in queue names to uppercase. You need a trailing colon when you include the queue name in an entry specification. Otherwise, the colon is optional.

Queue names must be unique, regardless of type. For example, if you define a print queue named SYSTEM, then you cannot also define a batch queue with the name SYSTEM.

9.1.1.3 Default Queues

You can define one default print queue and one default batch queue. When a user issues a PRINT or SUBMIT command and does not specify a queue name, PBS places the request on the appropriate default queue. You can change default queues at any time by using the SET QUEUE command. As distributed, PBS defines the default print queue SYS$PRINT and the default batch queue SYS$BATCH.

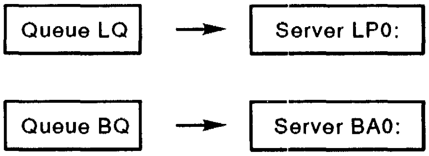
You do not have to define default queues. However, if you do not, a user must specify the queue on which to place each print or batch request.

9.1.1.4 Assigning Servers to Queues

Once you define a queue, users can submit requests to be placed on that queue. You must also indicate which servers are to process requests on that queue. You can define servers with the INITIALIZE/SERVER command.

In the simplest configuration, you assign each queue to a single server, and you do not assign two queues to the same server. Figure 9–1 shows this configuration.

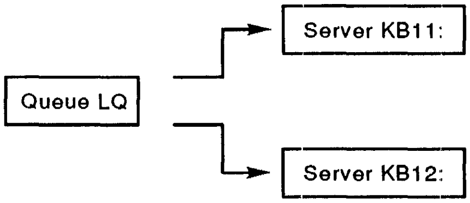
Figure 9–1: One-to-One Configuration



The one-to-one configuration is useful when users need to have their print requests printed on a particular printer or their batch request processed by a specific batch server. In such cases, you can choose the queue name to match its assigned server's name, such as print queue LP0: or batch queue BA0:.

It may be useful to assign more than one server to a queue, an option which is useful when multiple devices with the same characteristics exist. For example, at a site with two letter-quality printers (for example, KB11: and KB12:), you could define one print queue named LQ and assign it to servers controlling each device. Figure 9–2 shows this configuration.

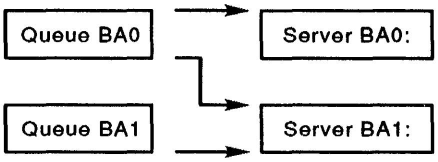
Figure 9–2: One-to-Many Configuration



The one-to-many configuration is useful when one class of devices can process similar jobs. When a user places a print request on the queue LQ, the first available server assigned to that queue processes the request. To balance the use of servers, PBS uses a round-robin method to determine which available server to use next. That is, in deciding which server to use next, the most recently used server is the last one PBS examines.

You can also assign more than one queue to the same server. For example, suppose a batch queue named BA0 is currently assigned to batch server BA0:, while batch queue BA1 is assigned to batch server BA1:. If you notice that queue BA0 has many entries waiting for processing while queue BA1 is empty, you can assign queue BA0 to both servers. Figure 9–3 shows this configuration.

Figure 9–3: Many-to-One Configuration



Note that in Figure 9–3 entries on queue BA1 are processed only by server BA1:, while entries on queue BA0 are processed by either server.

The combination of a one-to-many and a many-to-one configuration is a many-to-many configuration. That is, you can assign any set of queues to any set of servers. Thus, you have complete control over allocating resources (servers) for processing queue entries.

You can use the SHOW QUEUE/FULL command to display the servers assigned to a queue. You can use the SHOW SERVER/FULL command to display the queues assigned to a server. The ASSIGN/QUEUE and DEASSIGN/QUEUE commands let you assign and deassign each queue's servers.

9.1.1.5 Queue Priorities

You assign each queue a maximum priority and default priority in the range 1 to 255. The maximum priority of a queue defines the maximum priority a user can specify when issuing a request for that queue. The default priority of a queue defines the priority to assign to a request issued with no explicit priority specified. These two parameters let you set up queues so PBS processes all requests in one queue ahead of all requests in another queue.

Users with EXQTA privilege can submit PRINT or BATCH requests with a priority greater than the queue's maximum. See the section "Processing Requests" for more information on processing order.

9.1.1.6 Opening and Closing Queues

Normally, a queue accepts entries even if no servers are available to process the requests. At times, you may want to prevent PBS from placing any additional requests on a queue. This is called closing a queue; existing entries continue moving toward the front of the queue (if there are servers available to process them), but the queue does not accept additional entries. You can close a queue with the CLOSE/QUEUE command.

If a user issues a PRINT or SUBMIT command for a closed queue, PBS returns an error and rejects the request. You can open a closed queue with the OPEN/QUEUE command.

9.1.1.7 Starting and Stopping Queues

Entries move toward the head of a queue, and PBS processes them when they reach the queue's head. You can stop a queue with the STOP/QUEUE command to prevent any entries on the queue from being processed. Users can place additional entries on a stopped queue, but PBS does not process them. Any entries currently in process when you stop a queue are not affected. You can restart a stopped queue with the START/QUEUE command.

Note the distinction between stopping a queue and stopping a server assigned to a queue. When you stop a queue, PBS does not process any pending entries on that queue. When you stop a server, PBS does not process any pending entries on any queue assigned to that server. The two cases are similar in the one-to-one configuration. However, the two cases are quite different in a many-to-many configuration.

For example, if you assign both queues A and B to the same server, stopping queue A does not cause that server to become idle, since processing can still continue from queue B. However, if you stop the server, then processing stops on both queues.

9.1.1.8 Processing Requests

In the one-to-one or one-to-many configurations, PBS always processes the entry at the head of the queue. However, when you assign multiple queues to the same server, there is no implied order to processing entries; all of the entries at the front of each queue are, by definition, eligible for servicing. PBS resolves any conflicts that occur by checking the priority and age of the entries.

When several entries are waiting for processing, PBS chooses the one with the highest priority. If several entries have the same priority, then PBS selects the one with the earliest entry date and time.

Before PBS can process an entry, it must be in the READY state. This means that the entry:

- Is not in an AFTER state. An entry can be given an 'after' date and time value, which prevents the entry from being processed until the specified date and time has passed.
- Is not in a HOLD state. An entry can also be prevented from being processed until an operator or the entry's owner explicitly releases it.

Once an entry is in the READY state, PBS processes it as soon as all of the following conditions are met:

- PBS cannot process any other READY entries of the same type but with a higher priority
- PBS cannot process any other READY entries of the same type and the same priority but with an earlier entry date and time
- A server assigned to the entry's queue is available to process the entry. That is, the server is not currently processing any other entry.
- For print entries, a server's controlled device is not currently allocated or opened by any other job on the system.
- For print entries, at least one device is available with the entry's required forms installed on it.
- For batch entries, at least one pseudo keyboard of the proper type (static or dynamic, as appropriate to the server) is currently available to the server (that is, not allocated to or opened by another job on the system).

When all of the above conditions are met, PBS starts a print or batch job on behalf of the entry. PBS changes the entry's state to STARTED, and records the name of the server processing the job in the entry. Use the SHOW ENTRY/FULL command to display this information when an entry is in process.

Since PBS always processes print entries and batch entries by different servers, entries of a different type never compete with each other for processing.

9.1.1.9 Accessing Entries on a Queue

Users must also be able to locate entries in a queue. PBS provides two methods for accessing entries in queues:

- Entry number—When PBS adds an entry to a queue as a result of the PRINT or SUBMIT command, PBS assigns a unique entry number to it. Entry numbers are unique across all queues, regardless of type. Thus, each entry has a specific number assigned to it, whether it is in a print queue or a batch queue. PBS uses this number to record the entry in its queue file.
- Entry specification—You can also access an entry by identifying some or all of the following attributes:
  - The name of the entry's queue
  - The PPN of the entry's owner
  - The name of the entry

  These attributes are permanently established when you create the entry; they cannot be modified once PBS adds the entry to its queue.

  Several commands (for example, DELETE/ENTRY and SHOW ENTRY) let you identify an entry or group of entries by specifying an entry specification parameter with the command. The format of an entry specification is:

  queue-name:[PPN]entry-name

Unlike entry numbers, entry specifications do not uniquely identify an entry; several entries with the same attributes can exist at the same time. Therefore, be careful when using entry specifications with commands to modify or delete queue entries, since doing so can cause multiple entries to be modified or deleted inadvertently.

9.1.1.10 Job Limits for Queues

Each queue has maximum and default limits associated with it. For print jobs, you can define page limits; for batch jobs, you can define both CPU and elapsed time limits. The servers assigned to the queue enforce these limits.

This arrangement lets you set up queues that do not accept requests requiring many resources. For example, you could assign a low maximum page limit to a queue assigned to a server controlling a slow-speed printer, thus restricting requests to relatively small print jobs. Similarly, a queue assigned to a batch server running jobs at low priority could accept requests that include large CPU or elapsed time limits.

Users with EXQTA privilege can submit print or batch requests with job limits that exceed the maximum limits defined for the queues on which their requests are placed.

9.1.2 Print Servers

Each print server must have a unique name. PBS always names print servers after the devices they control (for example, server LP0: controls device _LP0:, server KB45: controls device _KB45:, and so on).

Some commands permit wildcard print server names. For example:

- *:—Means all servers
- LP*:—Means all LP-type servers
- KB*:—Means all KB-type servers

PBS does not allow the question mark character (ASCII 63) in print server names.

PBS associates each entry on a print queue with a form name. This name identifies the form that you must install on a printer before the server controlling that device can start the job. If you issue a PRINT command and do not specify a form name, PBS assigns the form set up as the default for the queue on which the request is placed.

9.1.2.1 Device Ownership

A print or batch server can retain ownership of its controlled device and prevent other jobs on the system from accessing the device. Use the /NOSHAREABLE qualifier with the INITIALIZE/SERVER and SET SERVER commands to indicate that the server's device should be permanently allocated to PBS. A device so allocated will be deallocated under any of the following conditions:

- An operator changes the server's NOSHAREABLE setting to SHAREABLE
- The device's server is deleted
- PBS is shut down

When PBS restarts after a shut down, it attempts to allocate any device whose server is marked as NOSHAREABLE. PBS continues to try allocating any device that is in use by another job on the system. PBS places such a server in the DEVICE-WAIT state indicating that the server is trying to gain access to the device.

If you initialize a server with NOSHAREABLE and its device is not available, PBS displays a warning message. PBS allocates the device as soon as it becomes available.

/SHAREABLE indicates that the device can be shared by other jobs on the system, and PBS allocates it only when the server starts processing a request. For a batch server, /SHARABLE also means the server uses a dynamic pseudo keyboard, while /NOSHARABLE means a static pseudo keyboard. The default is /SHAREABLE.

9.1.2.2 Job Copies and File Copies

Each print job consists of one or more files to be printed. Qualifiers are available with the PRINT command to specify the number of job or file copies to be printed.

The /JOB_COUNT=n command qualifier specifies the number of copies of the job to be printed. The /COPIES=n qualifier specifies the number of copies of a file to be printed. The default for both qualifiers is one copy.

If a print job consists of a single file, then the two qualifiers produce the same result except for the header pages (see the next section "Header Pages"). However, if a print job consists of more than one file, then the qualifiers produce different results.

For example, if you issue the following command:

```
$ PRINT/COPIES=2 FILE1.LST,FILE2.LST
```

the order of the job output is:

```
Job header (1 of 1)
File header (1 of 2)
FILE1.LST
File end
File header (2 of 2)
FILE1.LST
File end
File header (1 of 2)
FILE2.LST
File end
File header (2 of 2)
FILE2.LST
File end
```

But, if you issue the following command:

```
$ PRINT/JOB_COUNT=2 FILE1.LST,FILE2.LST
```

the order of the job output is:

```
Job header (1 of 2)
File header (1 of 1)
FILE1.LST
File end
File header (1 of 1)
FILE2.LST
File end
Job header (2 of 2)
File header (1 of 1)
FILE1.LST
File end
File header (1 of 1)
FILE2.LST
File end
```

9.1.2.3 Header Pages

You can print special job and file header pages to identify the start of a job or a file within a stack of listings. Header pages include large block letters to help you locate the beginning of job and file copies. You can control the number of job or file header pages printed by the JOB_PAGES=n and FLAG_PAGES=n attributes of the printer form.

Header pages consist of fields printed in single-height and double-height block letters and other information printed in six repeated lines. Three lines are printed at the top and bottom of each header page. These lines help you find the start of each job in a large listing.

The job header pages contain the following information:

- Owner's account name—This field is in single-height block letters
- Entry name—This field is in double-height block letters
- Owner's PPN—This field is in single-height block letters
- The three lines at the top and bottom of each job header page contain the following:
  - Current date
  - Current time

  - Entry entry-spec
  - Copy nnn of mmm
  - Printer device-name
  - Form form-name
  - Installation-name

The file header pages contain the following information:

- Device name and PPN—This field is in single-height block letters
- Filename—This field is in double-height block letters
- File type—This field is in double-height block letters
- Owner's PPN—This field is in single-height block letters
- The three lines at the top and bottom of each file header page contain the following:
  - Current date
  - Current time
  - File filespec
  - Copy nnn of mmm
  - Printer device-name
  - Form form-name
  - Installation-name

9.1.2.4 Page Limits

You can limit each print job to a specified maximum number of pages. The page limit can be:

- User specified
- The default for the queue
- Unlimited

Print servers count the number of pages they print. When a print job exceeds its page limit, PBS prints an error on the last page of the job listing and terminates the job. Note that header pages do not count toward a job page limit.

When the user specifies the page limit, the value can not exceed the maximum limit for the queue. If the user does not specify the page limit, PBS uses the default limit of the queue.

9.1.2.5 Control Characters and Escape Sequences

Some printers, such as the LN01, LN03, and LA2xx, can understand escape sequences. In some cases, the action taken in response to a particular escape sequence is device dependent. Therefore, you must route files that contain escape sequences to the correct server. This is best done by providing queues that assign jobs only to single servers or to servers that all serve the same type of printer.

The /CONTROLS qualifier, provided with the INITIALIZE/SERVER and SET SERVER commands, indicates how the server should process control characters and escape sequences:

- /CONTROLS—Indicates that the server sends control characters and escape sequences to its controlled device. The action taken is dependent on the device's handling of such characters. The print server does not update the form's current horizontal and vertical position when sending such characters to the printer.
- /CONTROLS=UP_ARROW—Indicates that the server translates control characters to an 'up arrow' format, consisting of a circumflex (^) character followed by a printable character that identifies the control character, and that the server translates the ESC character to the dollar sign ($), causing PBS to display all of the characters in the escape sequence. This is the default.
- /NOCONTROLS—Indicates that the server does not process control characters or escape sequences. PBS discards any such characters.

NOTE

Control characters and escape sequences can cause a print server to lose track of line and page positioning. This is because print servers do not know what action is taken by the printer in response to such characters. Normally, form feed characters correct positioning within a file unless the server is doing simulated form feed handling. In such cases, it may be necessary manually to adjust the printer alignment to correct positioning errors introduced by control characters and escape sequences.

DIGITAL recommends that servers processing control characters always use forms with the NOSIMULATE attribute, to prevent positioning errors between jobs.

9.1.2.6 Eight-Bit Characters

PBS passes eight-bit characters unchanged to printers designated as eight-bit devices with the SET PRINTER/EIGHT_BIT or SET TERMINAL/EIGHT_BIT commands.

For seven-bit devices, PBS converts eight-bit control characters to their seven-bit equivalents. For example, <CSI> becomes <ESC>[. PBS does not handle eight-bit printable characters; they are truncated to seven bits as they pass to seven-bit devices.

9.1.3 Batch Servers

Batch servers are named BAn:, where n is a unit number in the range 0-31. Some commands permit wildcard batch server names, for example:

- *:—Means all servers
- BA*:—Means all batch servers

PBS does not allow the question mark character (ASCII 63) in batch server names.

9.1.3.1 Executing Command Files

Batch servers use the DCL command file processor on pseudo keyboards to create the batch job environment. Batch servers execute a batch job by creating a temporary command file. This file:

- Opens a log file (unless the you specify /NOLOG_FILE with the SUBMIT command)
- Initializes error handling
- Executes LOGIN.COM files
- Turns VERIFY on
- Executes the individual user command files

PBS passes the parameters you specify with the SUBMIT command only to the first command file in the batch job.

Because batch servers create their own command procedure to execute the user's command files, the number of levels of command nesting that PBS allows is one less than the levels available when executing command procedures at the interactive level (see *RSTS/E Guide to Writing Command Procedures*).

The exit status that PBS reports in the batch log file is the status of the most severe error occurring in the job, not just the last one. The format of the exit status display is:

```
Exit status = <message>
```

Table 9–2 lists the possible exit status messages.

Table 9–2: PBS Batch Job Exit Status Messages

Message	Meaning
Entry deleted	Job was deleted during processing
Error	An error message occurred in the job, but no severe errors occurred
Exceeded CPU limit	Job exceeded the CPU time limit
Exceeded time limit	Job exceeded the time limit
Severe error	A severe error message occurred in the job
Success	No errors or warnings encountered
Unknown	Job logged out or was killed (exit status is unknown)
Warning	A warning message occurred in the job, but no errors occurred

9.1.3.2 CPU and Time Limits

Batch servers also monitor their batch jobs to enforce CPU and time limits and determine when the job is done. An active batch server examines its controlled job's CPU and time usage approximately every 30 seconds. If either value exceeds the allowable limit for the batch job, PBS terminates the job immediately. If you specify a log file with the SUBMIT command, PBS includes an error message in the log describing the reason for the job termination.

9.1.4 Managing Forms for PBS

Printer forms are the physical paper you install in a line printer. In some cases, forms are standard stock paper as normally used for printing program listings. I other cases, forms are customized, preprinted paper, such as checks or invoices.

PBS identifies each form by name. A form name consists of 1-9 characters from the set {A-Z, 0-9, $, _}. PBS converts lowercase characters to uppercase.

When you define a print server, you can specify the name for the form currently installed on the server's print device. Each form definition includes the form handling characteristics for the printer to which it applies. A Forms Definition File (FDF), maintains all the form characteristics for each server in PBS (see th next section).

When you issue a print request, you can specify the form name required using the /FORMS=form-name qualifier. If you do not include a form name, PBS assigns the queue's default form name to the print request. PBS verifies that th specified form name is in the FDF. If not, PBS returns an error and rejects the print request.

Before PBS can start a print request, a server assigned to the request's queue must be available with the required form installed on its printer. If not, the request remains READY until you change forms or until you assign a new serve with the correct form name to the request's queue.

You can define the same form name differently for different printers. For example, NORMAL can have different characteristics for the printer LP0: from those it has for LP1:. Generally, the physical dimensions of a form page (length and width) should remain the same from one printer to another, but the attributes that describe how a printer should handle the form may be different, depending on the device.

9.1.4.1 Forms Attributes

PBS defines the following attributes for printer forms:

- Form name—A unique 1 to 9 character name that identifies the form.
- Server name—The name of the server to which the form definition applies. This can refer to a single server or multiple servers of the same type (LP or KB).
- Length—The physical length of a form page, measured in lines of print. This value can be in the range 1 to 255.
- Width—The width of a form page, measured in characters printed across the page. The value can be in the range 1 to 255.
- Job pages—The number of header pages to print at the start of each new print job. The value can be in the range 0 to 127.
- Flag pages—The number of flag pages to print at the start of each new file within a print job. The value can be in the range 0 to 127.
- Form-feed handling—Depending on the form page length and the hardware capability, the printer may or may not be able to correctly handle FF characters that position the form page at the top of the next page. If not, then the server must simulate the form feed action by outputting the proper number of line feeds to reach the top of the next page.

- Bottom margin—The number of lines to skip at the bottom of each page. The value must be less than the form page length.
- Trailer pages—Determines whether or not a single trailer page prints per file.

9.1.4.2 The Forms Definition File

PBS provides a Forms Definition File (FDF) to maintain all printer form definitions. This file is a standard ASCII stream file, allowing anyone with write access to add, delete, and update the form definitions using a standard text editor. FDF defines form attributes by using keywords (for example, LENGTH=66), to simplify definitions further.

PBS treats lines beginning with an exclamation point (!) as comment lines. Thus, you can include descriptions in the definitions file.

PBS also supports continuation lines. If a definition ends with a hyphen (-), the definition is continued on the next line similar to the DCL command line continuation feature (see the *RSTS/E System User's Guide*).

NOTE

> If a line to be continued contains a comment, the hyphen must precede the comment (see the next section "Forms Definition Keywords").
>
> Blank lines are ignored. So, you can include blank lines where you want to separate definitions for readability.

The name of the Forms Definition File supplied on the RSTS/E distribution kit is PBS$:FORMS.SYS. As distributed, this file contains two definitions for the form named NORMAL: one is for LP-class printers and one is for KB-class printers. The file has a protection code of <40>, allowing any user on the system to read it; however, only users with WWRITE privilege can write to it.

The default print queue SYS$PRINT is initially set up with NORMAL as its default form name. After you install PBS, you can change any of these parameters. You can add additional definitions to the file as needed.

The order of form definitions in the FDF is significant. If you initialize a print queue and you do not include the /FORMS qualifier, PBS defaults to the first form definition. Similarly, if you define a print server and do not include the /FORMS qualifier, PBS defaults to the first form definition that applies to that server.

PBS accesses the FDF for the following reasons:

- When you include a /FORMS qualifier with the PRINT command, PBS searches the FDF for the specified form name.
- When you issue a INITIALIZE/SERVER or SET SERVER command that includes a /FORMS qualifier for a print server, PBS searches the FDF for the definition of the specified form name.
- When you issue a INITIALIZE/QUEUE or SET QUEUE command that includes a /FORMS qualifier for a print queue, PBS searches the FDF for the specified form name.

If the form name is not found, then PBS returns an error and rejects the request.

9.1.4.3 Forms Definition Keywords

A forms definition consists of:

- Form name
- Server name
- One or more keywords that describe the form attributes

You can use spaces or tabs to separate the keywords within a form definition. Any standard RSTS/E line terminator (CR, LF, FF) terminates each definition. For example, the definition of the default form named NORMAL for LP-class servers is:

```
NORMAL=_LP*:-      ! form NORMAL for all LP servers
LENGTH=66-         ! 66 lines per page
WIDTH=132-         ! 132 columns per line
JOB_PAGES=2-       ! 2 job header pages
FLAG_PAGES=2-      ! 2 flag header pages
BOTTOM_MARGIN=6-   ! 6 blank lines at the bottom of each page
TRAILER_PAGES-     ! trailer pages
NOSIMULATE         ! hardware top-of-form
```

If a forms definition applies to a particular server, or class of servers, the server name follows the form name and is separated from it by an equal sign. The format of the server name in a forms definition is:

form-name[=server-name[:]]

PBS does not permit embedded spaces or tabs between the form name and the server name.

The server-name identifies the server(s) to which this definition applies. If the definition only applies to a single server, then specify the explicit server name. If the definition applies to a class of servers (all LP or all KB devices), then specify form-name=_LP*: or form-name=_KB*: (wildcard unit number). If the definition applies to all print servers, then specify form-name=*: or omit this field from the definition. The default is all servers.

NOTE

> If a server belongs to more than one definition for the same form, place definitions that apply to specific servers before any that apply to a general class of servers. PBS searches the FDF sequentially until it finds the first occurrence of the requested form name with a matching server name.

You can abbreviate all forms definition keywords to their first two characters. Note that the form name always appears first in the definition and has no associated keyword. All other fields in the definition can be in any order. If a keyword is not included in a form definition, then PBS uses the keyword's default value.

You can include the following keywords in a forms definition:

LENGTH=n

Indicates the number of lines per printed page for the specified form. The value n can range from 1 to 255. The default length is 66; this is standard spacing for an 8-1/2 by 11-inch form with vertical spacing of 6 lines per inch.

NOTE

> Some devices that print on single-sheet forms cannot print on all physical lines of the form. For example, the LN01 prints a maximum of 60 lines per page. In such cases, the form length should be set to the device's maximum print lines; not the form's physical size.

WIDTH=n

Indicates the maximum number of characters that can appear in a printed line for the specified form. The value n can range from 1 to 255. The default width is 132. The WIDTH value determines where a printed line will be truncated (if the file is being printed with the /TRUNCATE qualifier of the DCL PRINT command in effect) or wrapped around to a new line (if the file is being printed with the /NOTRUNCATE qualifier in effect).

[NO]SIMULATE

Determines whether PBS should, on encountering a FF character or reaching the end of a page, send a FF character directly to the device (NOSIMULATE), or simulate the form feed by transmitting the proper number of line-feed characters (SIMULATE). The default is SIMULATE.

JOB_PAGES=n

Defines the number of job header pages to be printed at the beginning of each job listing. The value n can range from 0 to 127. Job header pages display information about the job (job name, PPN, and so forth), printed in large block letters. Job header pages make it easy to separate job listings. The default is JOB_PAGES=1.

FLAG_PAGES=n

Defines the number of file header pages to be printed at the beginning of each file listing. The value of n can range from 0 to 127. File header pages display information about the file (file name, PPN, and so forth), printed in large block letters. File header pages make it easy to separate printouts of files within a job. The default is FLAG_PAGES=1.

NOTE

> You can suppress the printing of file header pages by including the /NOFLAG_PAGES qualifier with the file to be printed.

BOTTOM_MARGIN=n

Defines the number of lines to skip at the bottom of each page of the form. The bottom margin value cannot exceed the form's length. The default is

BOTTOM_MARGIN=6. Printers that use single-sheet paper (for example, the LN01) automatically end a page and start a new page when it reaches its maximum print line. Forms definitions for these devices should normally specify BOTTOM_MARGIN=0.

NOTE

> You can suppress the printing of the bottom margin by including the /NOFEED qualifier with the file to be printed. This has no effect when BOTTOM_MARGIN=0.

TRAILER_PAGES
Determines whether or not a single trailer page prints after each file.

9.1.4.4 Changing Forms

Whenever you need to change forms on a printer under the control of PBS, perform the following steps:

1. Use the STOP/SERVER/JOB_END command to stop the server at the end of any current print job.
2. Use the SHOW SERVER command to confirm that the server is stopped and is not processing any print job.
3. Install the desired form in the printer.
4. Use the SET SERVER/FORMS=form-name command to assign a new form name to the server. Use the SHOW SERVER/FULL command to inspect the new form attributes (LENGTH, WIDTH, and so on) as needed.
5. Use the START/SERVER command to restart the server. When you issue this command, PBS routes any jobs that are waiting for the new form to be installed to the server.

Note that you can issue a SET SERVER/FORMS=form-name command without stopping the server, although DIGITAL does not recommended this unless the new form definition applies to the same physical form currently installed in the printer. Any print job in progress at the time its print server is assigned a new form name is not affected. The form name last assigned to the server determine the next job to be processed by the server. Use the SHOW SERVER command to display the name of the form most recently assigned.

If you modify the FDF to change the definition of a form name which is currentl assigned to a server, you must perform a form change (using the same form name to cause the new form definition to be read from the FDF.

9.2 Multiple Jobs

PBS consists of a single primary job and, optionally, one or more secondary jobs, depending on the number and use of concurrent print or batch servers. PBS is designed to provide a compact and efficient package across all of its jobs. This section provides some background information on how the PBS package makes use of multiple jobs.

9.2.1 The Primary Job

Whenever you issue the START/QUEUE/MANAGER command, PBS creates its primary job. This job runs detached. It contains the queue manager module and is always present when the package is running.

The primary job contains two print and two batch server slots. A slot refers to a sub-job in the PBS job that provides print or batch services. If the number of slots available to support the active print or batch servers in the primary job is insufficient, PBS creates secondary jobs.

9.2.2 Secondary Jobs

PBS creates secondary jobs when there are no print or batch server slots available in the primary job to process a waiting request. PBS attempts to activate only as many servers as necessary, depending on the number of waiting requests ready to run, available devices, and so on. Each secondary job contains three print server slots and four batch server slots.

When PBS starts, only the primary job is running. When a pending request is ready to start, the queue manager first determines if a slot is available on which to activate a server for the request. If the primary job contains no available server slots of the correct type (print or batch), then PBS automatically creates a secondary job.

Unlike the primary job, a secondary job remains present only while one or more of its server slots is allocated to a server. Once all the server slots in a secondary job become free (that is, there are no active servers in the job) PBS automatically removes the job from the system. It is possible for several PBS secondary jobs to be present on the system at the same time, depending on the number of active servers that you require at any time.

NOTE

> PBS secondary jobs contain only print and batch server slots; they do not contain a queue manager module.

Based on the configuration of PBS, the primary job is sufficient to handle two concurrent print jobs and two concurrent batch jobs. If more than two print or two batch servers are active at the same time, then PBS creates at least one secondary job. Each secondary job allows for three more print jobs and four more batch jobs to be started. Thus, if your printing volume causes six print devices to be active concurrently, you need a total of three RSTS/E jobs (one primary and two secondary) to run PBS.

PBS uses an "idle job timer" to determine when to remove a secondary job from the system. PBS removes a secondary job when all of its servers have been idle for five minutes. This time is sufficient for PBS to start a server's next job without having to recreate a secondary job.

PBS creates secondary jobs as needed, as long as job slots are available on the system. If none are available, then no secondary job is created, and PBS waits until either a job slot becomes available, or an existing server slot becomes free.

You can limit the number of secondary jobs required by controlling the number of servers that can process requests. You can do this by deassigning or stopping one or more servers. Idle servers that are stopped or not assigned to any queue cannot process a request. Thus, they do not require a server slot.

9.3 DCL Command Descriptions

The following sections contain descriptions of the PBS commands that require special privileges. See the *RSTS/E System User's Guide* for descriptions of PRINT, SUBMIT, DELETE/ENTRY, SET ENTRY, SHOW ENTRY, and SHOW QUEUE commands.

9.3.1 ASSIGN/QUEUE Command

The ASSIGN/QUEUE command assigns a server to a queue. Server assignments are cumulative; any server assigned to a queue is added to the set of assigned servers. A server named BAn: can only be assigned to a batch queue. Likewise, a server named LPn: or KBn: can only be assigned to a print queue. ASSIGN/QUEUE command requires PBSCTL privilege.

Format

ASSIGN/QUEUE *queue-name[:] server-name[:]*

Command Qualifiers	Defaults
none	none

Prompts

Queue: queue-name[:]
Server: server-name[:]

Command Parameters

queue-name[:]
Specifies the name of the queue to be assigned. This command does not permit wildcard queue names.

server-name[:]
Specifies the server to assign to the queue. This command does not permit wildcard server names.

9.3.2 CLOSE/QUEUE Command

This command closes a queue; that is, it prevents any additional requests from being placed on the queue. If the specified queue is not open, RSTS/E ignores this command. CLOSE/QUEUE requires PBSCTL privilege.

Format

CLOSE/QUEUE *queue-name[:]*

Command Qualifiers	**Defaults**
none	none

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be closed. This command does not permit wildcard queue names.

9.3.3 DEASSIGN/QUEUE Command

The DEASSIGN/QUEUE command deassigns a server from a queue. Any job already started on a deassigned server is not affected. However, no additional jobs are started on the server from the queue. DEASSIGN/QUEUE requires PBSCTL privilege.

Format

DEASSIGN/QUEUE *queue-name[:] server-name[:]*

Command Qualifiers	Defaults
/ALL	See Discussion

Prompts

Queue: queue-name[:]
Server: server-name[:]

Command Parameters

queue-name[:]
Specifies the name of the queue to be deassigned. This command does not permit wildcard queue names.

server-name[:]
Specifies the name of the server to be deassigned. This parameter overrides the /ALL qualifier. This command does not permit wildcard server names.

Command Qualifiers

/ALL
Indicates that all servers currently assigned to the queue should be deassigned. If you specify /ALL, PBS does not prompt for a server name. If you specify a server name, PBS ignores /ALL.

9.3.4 DELETE/QUEUE Command

The DELETE/QUEUE command deletes a queue. If you issue DELETE/QUEUE for a queue that has one or more entries, RSTS/E marks the queue for deletion and closes it. PBS deletes the queue as soon as it becomes empty. You can still display queues marked for deletion using the SHOW QUEUE command. DELETE/QUEUE automatically deassigns all servers from the deleted queue. DELETE/QUEUE requires SWCFG privilege.

Format

DELETE/QUEUE *queue-name[:]*

Command Qualifiers	Defaults
none	none

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be deleted. This command does not permit wildcard queue names.

9.3.5 DELETE/SERVER Command

This command deletes a print or batch server. If the server you specify is currently processing a job when you issue the command, RSTS/E marks it for deletion. PBS deletes the server when it becomes idle. DELETE/SERVER automatically deassigns the deleted server from all assigned queues. DELETE/SERVER requires SWCFG privilege.

Format

DELETE/SERVER *server-name[:]*

Command Qualifiers	**Defaults**
none	none

Prompts

Server: server-name[:]

Command Parameters

server-name[:]
Specifies the name of the server to be deleted. This command does not permit wildcard server names.

9.3.6 INITIALIZE/QUEUE Command

INITIALIZE/QUEUE defines a print or batch queue and establishes its initial properties. Use the SHOW QUEUE command to display the queues that are currently defined. See the *RSTS/E System User's Guide* for details on the SHOW QUEUE command.

When you initialize a queue, you also define its characteristics. Print and batch queues have different characteristics. You can modify any of a queue's characteristics later by using the SET QUEUE command.

Note that certain qualifiers for this command apply only to print queues, while others apply only to batch queues. An error results if you include both types of qualifiers in the same command. PBS uses the following criteria to determine whether a queue should be initialized as a print queue or a batch queue:

- If you give a /PRINT or /BATCH qualifier, it determines the queue's type.
- If you do not give either /PRINT or /BATCH, but you specify a print queue only qualifier (for example, /PAGE_LIMIT) or a batch queue only qualifier (for example, /CPU_LIMIT), that qualifier determines the queue's type.
- If you do not meet either of the previous conditions, the queue's type defaults to a print queue.

PBS retains queue information in the queue file when the package is shut down. The system uses this information to automatically reinitialize queues when it restarts the package. INITIALIZE/QUEUE requires SWCFG privilege.

Format

INITIALIZE/QUEUE *queue-name[:]*

Command Qualifiers	Defaults
/BATCH	See Discussion
/CLOSE	See Discussion
/CPU_LIMIT=(MAXIMUM=a,DEFAULT=b)	See Discussion
/DEFAULT	See Discussion
/FORMS[=form-name]	See Discussion
/PAGE_LIMIT=(MAXIMUM=a,DEFAULT=b)	See Discussion
/PRINT	See Discussion
/PRIORITY=(MAXIMUM=a,DEFAULT=b)	See Discussion
/PRIVILEGES=("priv[,...]")	/PRIVILEGES=NONE
/TIME_LIMIT=(MAXIMUM=a,DEFAULT=b)	See Discussion

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be initialized. This command does not permit wildcard queue names.

Command Qualifiers

/BATCH

Indicates that the queue is a batch queue, and only accepts entries with the SUBMIT command.

/CLOSE

Indicates that the queue should initially be closed, preventing any PRINT or SUBMIT entries from being entered on the queue. By default, PBS initializes the queue as open.

/CPU_LIMIT=(MAXIMUM=a,DEFAULT=b)
/CPU_LIMIT=n

Batch queues only. Indicates the maximum and default CPU time limits (in minutes) for each command file processed in a batch job started from this queue. The maximum CPU limit is the largest value that you can specify when issuing a request for the queue. The default CPU limit is the value that PBS assigns to a request that had no limit specified. The keyword UNLIMITED indicates that PBS does not enforce a CPU limit. If you do not specify the MAXIMUM argument, then PBS uses MAXIMUM=UNLIMITED. If you do not specify the DEFAULT argument, then PBS uses the maximum CPU limit as the default limit.

If you use the /CPU_LIMIT=n form, then PBS uses the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 65,535. The default CPU limit cannot exceed the maximum CPU limit.

If you issue a SUBMIT command without an explicit CPU limit, then PBS uses the queue default CPU limit. When you issue a SUBMIT command, you can specify a CPU limit up to, but not exceeding the queue maximum CPU limit.

/DEFAULT

Indicates that the queue should be marked as the default queue for its type (print or batch). You can define only one print queue and one batch queue as a default queue.

If you specify /DEFAULT, then the queue becomes the default queue for its type, and any queue currently defined as the default queue of the same type no longer serves as the default queue. By default, PBS does not mark the queue as a default queue.

/FORMS=form-name

Print queues only. Indicates if a default form name is defined for the queue. If you specify /FORMS=form-name, then PBS defines the name you specify as the default form name for the queue. If you issue a PRINT command without a /FORMS qualifier, then the queue's default form name is the first form name found in the FDF.

/PAGE_LIMIT=(MAXIMUM=a,DEFAULT=b)
/PAGE_LIMIT=n
Print queues only. Indicates the maximum and default page limits for each file printed as part of a print job started from this queue. The maximum page limit is the largest value that you can specify when issuing a PRINT request for this queue. The default page limit is the value assigned to a request for this queue that had no limit specified.

The keyword UNLIMITED indicates that no page limit is enforced. If you do not specify the MAXIMUM argument, then PBS uses MAXIMUM=UNLIMITED. If you do not specify the DEFAULT argument, then PBS uses the maximum page limit as the default limit.

If you use the /PAGE_LIMIT=n form, then PBS uses the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 4,294,967,295. The default page limit can not exceed the maximum page limit.

If you issue a PRINT command without an explicit page limit, then PBS uses the queue default page limit. When you issue a PRINT command, you can specify a page limit up to, but not exceeding, the maximum page limit of the queue.

/PRINT
Indicates that the queue is a print queue and only accepts entries from the PRINT command.

/PRIORITY=(MAXIMUM=a,DEFAULT=b)
/PRIORITY=n
Defines the maximum and default priorities for the queue, in the range 1 to 255. If you do not specify the MAXIMUM argument, then PBS uses 255. If you do not specify the DEFAULT argument, then PBS uses the maximum value as the default priority. If you use the /PRIORITY=n form, then PBS uses the value you specify for both the maximum and default priority. The default priority cannot exceed the maximum priority.

/PRIVILEGES=(priv,...)
Specifies the set of privileges required for a user to place entries on the queue. A user who issues a PRINT or SUBMIT command must have all of the queue's privileges for the request to be accepted. By default, a user does not need any privileges to place an entry on the queue.

The argument can be any of the RSTS/E privilege keywords, plus the keywords ALL and NONE. The /PRIVILEGES=ALL qualifier assigns all defined privileges to the queue; /PRIVILEGES=NONE assigns no privileges to the queue. Note that you can specify any privilege name with /PRIVILEGES, even those that you do not possess.

The argument can also be USER*n*, where *n* is an integer between 1 and 8. You can use these USER*n* privileges to identify particular users or groups of users and to restrict the queue's usage to those users.

/TIME_LIMIT=(MAXIMUM=a,DEFAULT=b)
/TIME_LIMIT=n
Batch queues only. Indicates the maximum and default elapsed time limits (in minutes) for each command file processed in a batch job started from this queue. The maximum time limit is the largest value that you can specify when issuing a request for this queue. The default time limit is the value that PBS assigns to a request that had no limit specified.

The keyword UNLIMITED indicates that PBS does not enforce a time limit. If you do not specify the MAXIMUM argument, then PBS uses MAXIMUM=UNLIMITED. If you do not specify the DEFAULT argument, then PBS uses the maximum time limit as the default limit. If you use the /TIME_LIMIT=n form, then PBS uses the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 65,535. The default time limit cannot exceed the maximum time limit of the queue.

If you issue a SUBMIT command without an explicit time limit, then PBS uses the queue default time limit. When you issue a SUBMIT command, you can specify a time limit up to, but not exceeding, the maximum time limit of the queue.

9.3.7 INITIALIZE/SERVER Command

The INITIALIZE/SERVER command defines a print or batch server. Use the SHOW SERVER command to display the servers that are currently defined. See the description of the SHOW SERVER command in this chapter for details.

When you initialize a server, you also define its characteristics. Print and batch servers have different characteristics. You can modify any of a server's characteristics later by using the SET SERVER command.

PBS retains the server information in the queue file when the package is shut down. The system uses this server information to initialize servers automatically when you restart the package. This eliminates the need to reinitialize servers each time you start PBS. INITIALIZE/SERVER requires SWCFG privilege.

Format

INITIALIZE/SERVER *server-name[:]*

Command Qualifiers	**Defaults**
/[NO]CONTROLS[=UP_ARROW]	See Discussion
/FORMS[=form-name]	See Discussion
/PRIORITY=n	See Discussion
/RUN_BURST=n	See Discussion
/[NO]SHAREABLE	/SHAREABLE

Prompts
Server: server-name[:]

Prompts

Command Parameters

server-name[:]
Specifies the name of the server to be initialized. This command does not permit wildcard server names.

Command Qualifiers

/[NO]CONTROLS[=UP_ARROW]
Print servers only. Indicates how print servers should process control characters and escape sequences. The default is /CONTROLS=UP_ARROW (see the section "Control Characters and Escape Sequences").

You must have PBSCTL privilege to specify this qualifier.

/FORMS[=form-name]
Print servers only. Identifies the form currently installed on the printer. The server only prints requests with this form name. PBS displays an error message if the form name does not exist in the FDF.

If you omit this qualifier, then PBS initializes the printer with the first form definition found in the FDF that applies to the specified print server. You can later issue a SET SERVER command to assign a different form to the server.

/PRIORITY=n

Batch servers only. Indicates the priority at which to run command procedures controlled by this server. RSTS/E permits any value in the range -120 to +120; however, batch jobs cannot exceed the priority of the PBS primary job. RSTS/E rounds the priority down to a multiple of 8. The default is -8.

You must have TUNE privilege to specify this qualifier.

/RUN_BURST=n

Batch servers only. Indicates the run burst to assign to any command procedures controlled by this server. RSTS/E permits any value in the range 1 to 127. The default is 6.

You must have the TUNE privilege to specify this qualifier.

/[NO]SHAREABLE

Specifies if the server's device should be shared with other jobs on the system. The default is /SHAREABLE.

If you specify /NOSHAREABLE, RSTS/E immediately allocates the device to the PBS job, thereby preventing other jobs on the system from gaining access to the device. For batch servers, PBS allocates the next available static pseudo keyboard to the server. The device remains allocated to the PBS job until you delete its server (with the DELETE/SERVER command), or you shut down PBS (with the STOP/QUEUE/MANAGER command). You get a warning message if you specify /NOSHAREABLE for a device currently allocated to another job on the system; PBS allocates the device when it becomes available.

If you specify /SHAREABLE, PBS allocates the device only when the server is active. The device is available to other jobs when the server is idle. The default is /SHAREABLE.

For batch servers, /SHAREABLE means the server uses a dynamic pseudo keyboard with terminal characteristics that can not be set; /NOSHAREABLE means the server uses a static pseudo keyboard, with terminal characteristics that can be set. DIGITAL recommends that you use /SHAREABLE unless you need specific terminal characteristics

9.3.8 OPEN/QUEUE Command

The OPEN/QUEUE command opens a queue for additional requests after it has been closed. If the specified queue is already open, RSTS/E ignores this command OPEN/QUEUE requires PBSCTL privilege.

Format

OPEN/QUEUE *queue-name[:]*

Command Qualifiers	**Defaults**
none	none

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be opened. This command does not permit wildcard queue names.

9.3.9 SET QUEUE Command

The SET QUEUE command modifies the characteristics of one or more queues. Changing a queue's characteristics does not affect entries already on the queue. SET QUEUE requires PBSCTL privilege.

Format

SET QUEUE *queue-name[:]*

Command Qualifiers	Defaults
/ALL	See Discussion
/BATCH	See Discussion
/CPU_LIMIT=(MAXIMUM=a,DEFAULT=b)	Attributes are unchanged
/[NO]DEFAULT	Attributes are unchanged
/FORMS[=form-name]	Attributes are unchanged
/PAGE_LIMIT=(MAXIMUM=a,DEFAULT=b)	Attributes are unchanged
/PRINT	See Discussion
/PRIORITY=(MAXIMUM=a,DEFAULT=b)	Attributes are unchanged
/PRIVILEGES=("priv[,...]")	Attributes are unchanged
/TIME_LIMIT=(MAXIMUM=a,DEFAULT=b)	Attributes are unchanged

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be modified. This command permits wildcard queue names. The rules for selecting queues when using wildcards are:

- If you specify /PRINT or a print queue only qualifier, then PBS only selects print queues.
- If you specify /BATCH or a batch queue only qualifier, then PBS only selects batch queues.
- If neither of the above rules are true, then PBS selects any queue.

Command Qualifiers

/ALL

Indicates that all queues should be modified. If you specify /ALL, PBS omits the queue name prompt and uses the queue name *. If you specify a queue name, PBS ignores this qualifier.

/BATCH

Indicates that only batch queues should be modified. When you use /ALL with this qualifier, PBS modifies all batch queues.

/CPU_LIMIT=(MAXIMUM=a,DEFAULT=b)
/CPU_LIMIT=n

Batch queues only. Indicates the maximum and default CPU time limits (in minutes) for each command file processed in a batch job started from this queue. The maximum CPU limit is the largest value that you can specify when issuing request for the queue. The default CPU limit is the value that PBS assigns to a request that had no limit specified.

The keyword UNLIMITED indicates that PBS does not enforce a CPU limit. If you do not specify the MAXIMUM argument, then PBS does not change the maximum CPU limit. If you do not specify the DEFAULT argument, then PBS does not change the default CPU limit. If you use the /CPU_LIMIT=n form, ther PBS use the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 65,535. The default CPU limit cannot exceed the maximum CPU limit.

If you do not specify this qualifier, then PBS does not change the queue's curren default and maximum values.

/[NO]DEFAULT

Indicates whether the queue should serve as the default queue for its type. If you specify /DEFAULT, then PBS marks the queue as the default queue for its type, and any queue already defined as the default queue is no longer marked as being the default queue. If you specify /NODEFAULT, then the queue is no longer marked as being the default queue for its type. You then have no default queue for that type. PBS does not return an error if you specify /NODEFAULT for a queue that is not currently marked as the default queue.

If you do not specify this qualifier, then any current default queue is not affected

/FORMS=form-name

Print queues only. Indicates if a default form name is defined for the queue. If you specify /FORMS=form-name, then the name you specify becomes the default for the queue and replaces any previous default form.

If you do not specify this qualifier, then the queue's current default form name i not changed.

/PAGE_LIMIT=(MAXIMUM=a,DEFAULT=b)
/PAGE_LIMIT=n

Print queues only. Indicates the maximum and default page limit for each file printed as part of a print job started from this queue. The maximum page limit is the largest value that you can specify when issuing a request for this queue. The default page limit is the value that PBS assigns to a request for this queue that had no limit specified.

The keyword UNLIMITED indicates that no page limit is enforced. If you do no specify the MAXIMUM argument, then PBS does not change the maximum pag limit. If you do not specify the DEFAULT argument, then PBS does not change the default page limit. If you use the /PAGE_LIMIT=n form, then PBS uses the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 4,294,967,295. The default page limit can not exceed the maximum page limit.

If you do not specify this qualifier, then the queue's current default and maximu values are not changed.

/PRINT

Indicates that only print queues should be modified. When you use /ALL with this qualifier, PBS modifies all print queues.

/PRIORITY=(MAXIMUM=a,DEFAULT=b)
/PRIORITY=n

Defines the maximum and default priorities for the queue, in the range 1 to 255. If you do not specify the MAXIMUM argument, then PBS does not change the current maximum priority. If you do not specify the DEFAULT argument, then PBS does not change the current default priority. If you use the /PRIORITY=n form, then PBS uses the value you specify for both the maximum and default priorities. The default priority cannot exceed the maximum priority.

If you do not specify this qualifier, then the queue's current default and maximum values are not changed.

/PRIVILEGES=(priv,...)

Specifies the set of privileges required for a user to place entries on the queue. A user who issues a PRINT or SUBMIT command must have all of the queue's privileges for the request to be accepted. PBS accepts any of the RSTS/E keywords, plus the keywords ALL and NONE.

The /PRIVILEGES=ALL qualifier assigns all defined privileges to the queue; /PRIVILEGES=NONE assigns no privileges to the queue. Note that you can specify any privilege name with /PRIVILEGES, even those that you do not possess.

The argument can also be USER*n*, where *n* is an integer between 1 and 8. You can use these USER*n* privileges to identify particular users or groups of users and to restrict the queues usage to those users.

If you specify this qualifier, PBS replaces the queue's current privileges with those in the argument list. If you do not specify this qualifier, the queue's current privileges are unchanged.

/TIME_LIMIT=(MAXIMUM=a,DEFAULT=b)
/TIME_LIMIT=n

Batch queues only. Indicates the maximum and default elapsed time limits (in minutes) for each command file processed in a batch job started from this queue. The maximum time limit is the largest value that a user can specify when issuing a request for this queue. The default time limit is the value PBS assigns to a request that had no limit specified.

The keyword UNLIMITED indicates that PBS does not enforce a time limit. If you do not specify the MAXIMUM argument, then PBS does not change the current maximum time limit. If you do not specify the DEFAULT argument, then PBS does not change the current default time limit. If you use the /TIME_LIMIT=n form, then PBS use the value you specify for both the maximum and default limits. If you specify a numeric argument, it must be in the range 1 to 65,535. The default time limit cannot exceed the maximum time limit.

If you do not specify this qualifier, then the queue's current default and maximum values (if any) are not changed.

9.3.10 SET SERVER Command

The SET SERVER command changes the characteristics of one or more servers. Use this command to change forms for print servers. SET SERVER requires PBSCTL privilege.

Format

SET SERVER *server-name[:]*

Command Qualifiers	Defaults
/ALL	See Discussion
/BATCH	See Discussion
/[NO]CONTROLS[=UP_ARROW]	Attributes are unchanged
/FORMS[=form-name]	Attributes are unchanged
/PRINT	See Discussion
/PRIORITY[=n]	Attributes are unchanged
/RUN_BURST[=n]	Attributes are unchanged
/[NO]SHAREABLE	Attributes are unchanged

Prompts

Server: server-name[:]

Command Parameters

server-name[:]

Specifies the name of the server to be modified. This command permits wildcard server names.

Command Qualifiers

/ALL

Indicates that all servers should be modified. If you specify /ALL, PBS omits the server name prompt. If you specify a server name, PBS ignores this qualifier.

/BATCH

Indicates that only batch servers should be modified. When you use /ALL with this qualifier, PBS modifies all batch servers.

/[NO]CONTROLS[=UP_ARROW]

Print servers only. Indicates how print servers should process control characters and escape sequences. If you do not specify this qualifier, then PBS does not change the current CONTROLS setting. You must have the PBSCTL privilege to use this qualifier.

/FORMS=form-name

Print servers only. Indicates the name of the form installed on the printer. The server only prints requests with the same form name. PBS returns an error if the form name you specify for the device does not exist in the FDF.

If you do not specify this qualifier, then the server's current form is not changed.

/PRINT

Indicates that only print servers are to be modified. When you use /ALL with this qualifier, PBS modifies all print servers.

/PRIORITY[=n]

Batch servers only. Indicates the priority at which to run command procedures controlled by this server. This value cannot exceed the priority of the PBS primary job. PBS accepts any value in the range -120 to +120. PBS rounds the value down to a multiple of 8.

If you do not specify an argument with /PRIORITY, the server's priority is set to -8. If you do not specify this qualifier, then the batch server's priority is not changed. You must have the TUNE privilege to use this qualifier.

/RUN_BURST[=n]

Batch servers only. Indicates the run burst to assign to any command procedures controlled by this server. PBS accepts any value in the range 1 to 127.

If you do not specify an argument with /RUN_BURST, the server's run burst is set to 6. If you do not specify this qualifier, then the batch server's run burst is not changed. You must have the TUNE privilege to use this qualifier.

/[NO]SHAREABLE

Indicates if the device should be made shareable with other jobs on the system.

If you specify /NOSHAREABLE, then PBS immediately allocates the device, preventing other jobs from accessing the device. The device is deallocated whenever you delete the server (with the DELETE/SERVER command) or shut down PBS (with the STOP/QUEUE/MANAGER command). If you specify /NOSHAREABLE and the device is not currently available, PBS displays a warning message and allocates the device as soon as it becomes available.

/SHAREABLE indicates that the device can be shared by other jobs on the system and PBS only allocates it when the server is active.

For batch servers, /SHAREABLE means the server uses a dynamic pseudo keyboard with terminal characteristics that can not be set; /NOSHAREABLE means the server uses a static pseudo keyboard, with terminal characteristics that can be set. DIGITAL recommends that you use /SHAREABLE unless you need specific terminal characteristics

If you do not specify this qualifier, then the [NO]SHAREABLE characteristic is not changed.

9.3.11 SHOW SERVER Command

The SHOW SERVER command displays the status of one or more servers in PBS. The display includes information about the server's characteristics, its current status, and the queues assigned to the server. The SHOW SERVER command requires no privileges.

Format

SHOW SERVER *[server-name[:]]*

Command Qualifiers	Defaults
/ALL	/ALL
/BATCH	none
/BRIEF	/BRIEF
/FULL	/BRIEF
/PRINT	none

Prompts

none

Command Parameters

server-name[:]

Specifies the name of the server to be displayed. If you do not specify a server name, all servers are displayed. This command does not permit wildcard server names.

Command Qualifiers

/ALL

Specifies that all servers are to be displayed. /ALL is the default if no server name is specified. If a server name is specified, this qualifier is ignored.

/BATCH

Specifies that only batch servers are to be displayed.

/BRIEF

Specifies a brief display. It is the default. The brief display includes the following information:

- On-line or Off-line—Print servers only. Indicates the current status of the server's print device. Note that PBS does not detect an off-line device until it attempts to print on it; all devices are assumed to be on-line when PBS first starts. KB-type printers are considered off-line when an XOFF (Ctrl/S) characters causes the device to stall for an extended period of time.
- Active—Indicates that the server is currently processing a request.
- Idle—Indicates that the server is not currently processing a request.
- Stopped—Indicates that the server is currently stopped and will not accept any new requests. The server may or may not be active.

- Job-wait—Indicates that PBS is unable to create a secondary job because no job slots are currently available. PBS will keep trying to create the second job until it succeeds.
- Device-wait—Indicates that PBS is unable to allocate a device because the device is not available. For print servers, device-wait means that the printer controlled by the server is allocated to or opened by another job. For batch servers, device-wait means that no pseudo-keyboards are currently available. PBS will keep trying to allocate the device until it succeeds.
- Shareable or Non-shareable—For print servers, shareable means that the print device is available to other jobs on the system when it is not in use by the server. Non-shareable means that the print device is permanently allocated to the server.

  For batch servers, shareable means the server uses a dynamic pseudo keyboard, with non-specifiable characteristics. Non-shareable means that the server uses a static pseudo keyboard that therefore does not change characteristics; rather, it uses the characteristics specified.
- Form name—Print servers only. Indicates the name of the forms currently installed, or "No forms" if no forms are installed.

/FULL

Specifies a full display. The full display includes all of the information in the brief display, and also includes the following information:

- Current entry—If the server is active, PBS displays the entry number and entry spec of the current entry.
- Assigned queues—PBS displays a list of the queues which can route requests to the server.

/PRINT

Specifies that only printer servers are to be displayed.

9.3.12 START/QUEUE Command

The START/QUEUE command starts a queue that was stopped with the STOP/QUEUE command. Once you start a queue, PBS processes waiting entries on the queue. If you already started the specified queue, PBS ignores this command. START/QUEUE requires PBSCTL privilege.

Format

START/QUEUE *queue-name[:]*

Command Qualifiers	Defaults
none	none

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]

Specifies the name of the queue to be started. This command does not permit wildcard queue names.

9.3.13 START/QUEUE/MANAGER Command

The START/QUEUE/MANAGER command starts PBS. Following this procedure, it:

1. Creates the PBS primary job as a detached job running in the user's account. RSTS/E initializes the job's priority and run burst according to the package configuration parameters.
2. Compresses the PBS system file by removing any records that are marked as deleted. (This step reduces the overall size of the file.) During this step, if RSTS/E finds an entry with status STARTED, it retains the entry, placing it on HOLD. This can happen if RSTS/E restarts the package after a system crash.
3. Loads all queue definitions into memory from the PBS system file.
4. Loads all server definitions into memory from the PBS system file. If a print server device is not a valid device on the system, PBS marks the server DISABLED.
5. Assigns servers to queues based on the assignment information maintained in the PBS system file.
6. Attempts to permanently allocate a device for any server marked as NOSHAREABLE in its server's definition. If the allocation fails because the device is currently in use, PBS marks the server DEVICE-WAIT and attempts to allocate the device periodically until the allocation succeeds.
7. Confirms startup has successfully completed by displaying a message.

The start-up process can take from a few seconds to several minutes, depending on the number of pending requests in the queue. Most users include the START/QUEUE/MANAGER command in their system start-up command file so the time required for PBS start-up is added to normal system start-up time. START/QUEUE/MANAGER requires PBSCTL privilege.

Format

START/QUEUE/MANAGER

Command Qualifiers	Defaults
/PRIORITY=n	/PRIORITY=-8
/RUN_BURST=n	/RUN_BURST=6

Prompts

none

Command Qualifiers

/PRIORITY=n

Specifies the priority of the detached PBS job and any secondary jobs that PBS creates. Value must be in the range -120 to +120; with any value specified rounded down to a multiple of 8. The default is -8.

/RUN_BURST=n

Specifies the run burst of the detached PBS job and any secondary jobs that PBS creates. Value must be in the range 1 to 127. The default is 6.

9.3.14 START/SERVER Command

This command restarts a server that was stopped with the STOP/SERVER command. If the server has a stalled job when you issue the command, the command either resumes the job or terminates it, depending on the qualifiers you use.

For print servers, the job can be resumed at a specified page of the current file copy. PBS ignores this command if the specified server is already started. For batch servers, PBS releases the controlled batch job from its held state. START/SERVER requires PBSCTL privilege.

Format

START/SERVER *server-name[:]*

Command Qualifiers	Defaults
/BACKSPACE[=n]	See Discussion
/FORWARDSPACE[=n]	See Discussion
/NEXT_JOB	See Discussion
/PAGE=n	See Discussion
/RESTART	See Discussion
/TOP_OF_FILE	See Discussion

Prompts

Server: server-name[:]

Command Parameters

server-name[:]

Specifies the name of the server to be started. This command does not permit wildcard server names.

Command Qualifiers

/BACKSPACE[=n]

Print servers only. Indicates that printing should resume n pages back in the current file listing. If you do not include an argument, then printing resumes one page before the current page. If you specify an argument that extends beyond the beginning of the file, then printing resumes at the beginning of the current file copy.

PBS ignores this qualifier if the specified server is idle.

/FORWARDSPACE[=n]

Print servers only. Indicates that printing should resume n pages forward in the current file listing. If you do not include an argument, printing resumes one page after the current page. If you specify an argument that extends beyond the end of the current file, printing resumes at the next file copy, if any.

PBS ignores this qualifier if the specified server is idle.

/NEXT_JOB

Indicates that the current job should be aborted and printing or batch processing should resume at the next job, if any.

PBS ignores this qualifier if the specified server is idle.

/PAGE=n

Print servers only. Indicates that printing should resume at page n in the current file listing. If you specify an argument that extends beyond the end of the file, printing resumes at the next file copy, if any. A value of 1 indicates the first page of the file; PBS does not include header pages in the count.

PBS ignores this qualifier if the specified server is idle.

/RESTART

Print servers only. Indicates that printing should resume at the beginning of the current job copy. If you submit the job with a /JOB_COUNT qualifier argument larger than one, PBS only restarts the current job copy.

PBS ignores this qualifier if the specified server is idle.

/TOP_OF_FILE

Print servers only. Indicates that printing should resume at the beginning of the current file copy. If the file being printed was submitted with a /COPIES qualifier argument larger than one, PBS only restarts the current file copy.

PBS ignores this qualifier if the specified server is idle.

9.3.15 STOP/QUEUE Command

The STOP/QUEUE command stops a queue; that is, PBS prevents any pending requests on the queue from being processed. Any requests that are currently being processed are not affected. STOP/QUEUE requires PBSCTL privilege.

PBS ignores this command if the specified queue is already stopped.

Format

STOP/QUEUE *queue-name[:]*

Command Qualifiers	Defaults
none	none

Prompts

Queue: queue-name[:]

Command Parameters

queue-name[:]
Specifies the name of the queue to be stopped. This command does not permit wildcard queue names.

9.3.16 STOP/QUEUE/MANAGER Command

The STOP/QUEUE/MANAGER command lets you shut down PBS, either immediately (causing all currently active jobs to be terminated) or when all servers become idle. STOP/QUEUE/MANAGER requires PBSCTL privilege.

You can shut down PBS in two ways:

- As part of normal system shutdown. The SHUTUP program includes a phas for automatically shutting down PBS, either immediately or when all print o batch jobs currently in progress complete.
- Manually, using the STOP/QUEUE/MANAGER command.

Format

STOP/QUEUE/MANAGER

Command Qualifiers	Defaults
/[NO]ABORT	/NOABORT

Prompts

none

Command Qualifiers

/[NO]ABORT

Indicates whether any print or batch jobs currently running should be aborted before shutting down PBS. Use /ABORT to shut down PBS immediately; use /NOABORT to let all currently running print or batch jobs complete. The default is /NOABORT.

9.3.17 STOP/SERVER Command

This command stops a print or batch server. The action taken depends on whether the server is currently active or not. For queues that are assigned to only one server, stopping the server is equivalent to stopping the queue. For queues that are assigned to more than one server, stopping one server does not prevent requests from being routed to other active servers. STOP/SERVER requires PBSCTL privilege.

If you issue a STOP/SERVER command for a server that is not currently processing a request, no additional requests are routed to that server. However, if you issue a STOP/SERVER command for a server that is currently processing a request, the following action is taken depending on the server:

- Batch servers—If you specify the /JOB_END qualifier, PBS lets the current batch job complete before stopping the server. Otherwise, the server suspends the batch job by assigning it a priority of -128. The server then waits until PBS issues a START/SERVER command to either resume or terminate the batch job.
- Print servers—If you specify the /PAGE_END, /FILE_END or /JOB_END qualifier, PBS does not stop the server until it completes the current page, file copy, or job copy. If the copy is the final copy of the job, PBS lets the job finish before it stops the server. If you do not specify any of the qualifiers, PBS stops the server immediately.

  When PBS directs the server to stop, the server stops sending data to the printer. However, system buffers continue to be printed on the device until they become empty. This is necessary to prevent PBS from losing proper page positioning on the printer. The server then waits until you issue a START/SERVER command to either resume or terminate the print job.

Format

STOP/SERVER *server-name[:]*

Command Qualifiers	Defaults
/FILE_END	See Discussion
/JOB_END	See Discussion
/PAGE_END	See Discussion

Prompts

Server: server-name[:]

Command Parameters

server-name[:]

Identifies the server to be stopped. This command does not permit wildcard server names.

Command Qualifiers

/FILE_END

Print servers only. Indicates that printing should stop after the current file copy is finished printing.

Note that this qualifier conflicts with /JOB_END and /PAGE_END.

/JOB_END

For print servers, indicates that printing should stop after the device has finished printing the current job. For batch servers, indicates that the server should stop when the current batch job completes.

Note that this qualifier conflicts with /FILE_END and /PAGE_END.

/PAGE_END

Print servers only. Indicates that printing should stop at the end of the current page.

Note that this qualifier conflicts with /FILE_END and /JOB_END.

Chapter 10

Managing System Errors

The RSTS/E monitor automatically logs both hardware and system level software errors. To gain the full advantages of this error detection capability, you must use the programs in the System Error Package. These programs fall into three main categories:

- Extraction and retention of system errors
- Extraction and retention of system errors not yet retrieved at the time of a system crash
- Compilation and formatting of saved system errors

When the monitor detects an error, routines save critical error-related data and send a message to the ERRCPY program. The system invokes ERRCPY, which then retrieves the saved data, performs minimal error message processing, and stores the error information in a specially formatted disk file (ERRLOG.FIL). Because RSTS/E limits the number of messages that may be queued to ERRCPY, it should be run at a higher priority than most jobs to prevent the loss of valuable diagnostic information. A separate program, ERRINT, initializes and validates the error logging file to minimize the size of ERRCPY. Normally, you run ERRINT at the beginning of system start-up to start ERRCPY.

When a system crash occurs and the system crash dump facility is enabled, the monitor preserves the contents of certain critical parts of the system in the system file [0,1] CRASH.SYS. Run the ANALYS program immediately following the recovery from a system crash to extract and format key information from the crash file. Normally, this is done by the system start-up command file. One of the functions ANALYS performs is the creation of a separate error logging file (by default, ERRCRS.FIL) containing errors that were not processed by ERRCPY at the time of the crash and an ERRDIS report for the errors.

NOTE

> You must have SWCTL privilege to run any component of the System Error Package.

10.1 Use of the Error Logging Programs: ERRINT and ERRCPY

The system program ERRCPY reads error-related information stored in the monitor part of memory and writes it to a special disk file called ERRLOG.FIL. The error file initialization and validation program (ERRINT) starts the ERRCPY program. ERRINT performs various file checking or generation functions and chains to ERRCPY. You must make sure the commands are in the START.COM file to start ERRINT (thus, ERRCPY) and that ERRCPY is active during time-sharing operations.

10.1.1 Error Logging Initialization: ERRINT

When the RSTS/E system starts up, DCL executes commands in the START.COM command file. The following command starts ERRINT:

```
$ RUN ERROR$:$ERRINT
```

When the system executes the previous command, the ERRINT program tries to locate the error file ERRLOG.FIL, which resides in ERROR$:. If it finds the file, ERRINT checks certain critical control information in the file. If this information is invalid, ERRINT renames the file ERRLOG.TMP, generates and initializes a new ERRLOG.FIL, and issues an informational message:

```
Error File was found to be Invalid
Error File was generated and Invalid File renamed ERRLOG.TMP
```

If ERRINT does not find ERRLOG.FIL, it generates and initializes the file.

Two other commands, corresponding to responses to ERRINT questions, must also be present in the command file. The two ERRINT dialogue questions for which you must include responses are:

```
Change size to < 100 >?
Utilize crash file output (YES/NO) <NO>?
```

Once ERRINT validates or creates ERRLOG.FIL, it prints a message telling you what percentage of the file has been used. For example:

```
ERRLOG file is 8% full
```

ERRINT also prints a question that includes (in angle brackets) the maximum number of blocks allowed for the file:

```
Change size to < 100 >?
```

You can accept the current size of the file (originally 100 blocks) or change the maximum size if you want. You can also specify an optional switch, /LOCK. If you include /LOCK in your response, ERRCPY runs nonswappable. The default is that ERRCPY can be swapped out.

NOTE

> When an error occurs, RSTS/E writes it to a disk file. If ERRINT has been swapped out to this same disk, RSTS/E does not log the error. However, if you lock ERRINT in memory, RSTS/E does not lose these errors. This is the only advantage of locking ERRINT in memory.
>
> The disadvantage of locking ERRINT in memory is that the system may hang, particularly if the memory is fragmented. The probability of the system hanging for this reason is a function of free memory available at any given time. Because the free memory is, in part, a function of total available memory, the probability of your system hanging for this reason decreases with increased total memory.

The program then asks if you want to append the contents of the special error crash file ERRCRS.FIL to the contents of ERRLOG.FIL in the question:

```
Utilize crash file output (YES/NO) <NO>?
```

If you answer YES, ERRINT tries to find the file and add its contents to the end of the main error logging file. If it cannot find the error crash file, ERRINT ignores the command and continues.

After it finishes processing all of its functions, ERRINT detaches and chains to ERRCPY. At this point, the terminal on which ERRINT started is free for other use.

The ERRINT program must be located in the error library account ERROR$. To run ERRINT from this account, type RUN ERROR$:ERRINT and press Return.

You can run ERRINT only if ERRCPY is not currently running. If you start up ERRINT while ERRCPY is running, then ERRINT exits after displaying the following message:

```
?Error logger is already running
```

Note that the version of ERRCPY that was running in a detached state on the system continues to run.

10.1.2 Examples of ERRINT Dialogue

An example of the ERRINT dialogue follows:

```
$ run error$:errint<RET>
ERRINT V9.7  RSTS V9.7  EDERE
ERRLOG file is 8% full
Change size to < 100 >? 120<RET>
Utilize crash file output (YES/NO) <NO>? NO<RET>
Detaching
```

In this example, a valid error log file exists, has a maximum length of 100 blocks, and is 8% full. If you change the maximum size to 120 blocks but decide not to add the special ANALYS error log file to the ERRINT error log file, the program detaches and chains to ERRCPY.

In the next example, ERRINT finds an invalid error log file, renames the invalid file to ERRLOG.TMP, creates a new file, leaves the maximum size at 100 blocks, and attaches the special error log file ERRCRS.FIL to the error log file ERRLOG.FIL. Note that the 2% full message indicates a file containing only control information; this is the state of a newly generated error log file:

```
$ run error$:errint<RET>
ERRINT V9.7  RSTS V9.7  EDERE
Error file was found to be invalid
New file was generated and invalid file renamed ERRLOG.TMP
ERRLOG file is 2% full
Change size to < 100 >?<RET>
Utilize crash file output (YES/NO) <NO>? YES<RET>
Detaching
```

10.1.3 Error Logging: ERRCPY

You can start the ERRCPY program only by running ERRINT. ERRCPY runs detached, processes incoming error messages, and writes each message as a variable-length record to a nonspanned sequential file (ERRLOG.FIL). When its message queue is empty, ERRCPY enters the receiver sleep state and may be swapped out.

The monitor queues messages to ERRCPY regardless of the presence or absence of ERRCPY. (If ERRCPY is not running, the messages are simply left waiting, up to a limit of 40.) The first message the monitor queues on start-up is always a "power fail/start up" message. On receiving a special error message from the SHUTUP program, ERRCPY logs the message, closes the error file, and kills itself.

10.2 Displaying Errors: ERRDIS

The error display program ERRDIS provides you with a convenient method of displaying previously logged errors.

ERRDIS has four functions:

- Provides a summary report by error type and unit number (where applicable) of all errors logged to the error file
- Provides a detailed report of one or all error types logged between any two user-selected date and time pairs
- Zeros some or all of the contents of the error file following the generation of a report
- Provides a list of potentially bad disk blocks

ERRDIS consists of two modules: ERRDIS and ERRDET. The first module performs the first and third functions. To perform the other functions, ERRDIS chains to ERRDET; control returns to ERRDIS when ERRDET finishes processing.

10.2.1 Running ERRDIS

You can run the ERRDIS program by typing:

```
$ RUN ERROR$:ERRDIS
```

After ERRDIS prints header information, you select optional modes of operation by means of an interactive dialogue. In the dialogue description shown in Table 10–1, prompts are numbered for reference. An answer enclosed in angle brackets (<>) indicates the default response; select the default by pressing Return in response to a program prompt. You can abbreviate most responses with two characters. Optional characters are enclosed in square brackets [ ]; for example, HE[LP].

Table 10–1: ERRDIS Dialogue Explanation

Step	Prompt and Description
1	ERRDIS V9.7 RSTS V9.7 EDERE ERRDIS prints an identification line.
2	Input File <ERROR$:ERRLOG.FIL>? Enter a file specification for ERRDIS to process. The file must be formatted as an error logging file.
3	Output to <KB:ERRDIS.OUT>? Enter a file specification to which ERRDIS should send the report.
4	He[lp], Ba[d Blocks], Su[mmary], Fu[ll] Report, or Ze[ro] File? <Summary>? Select one of five possible reports: • He[lp]. Prints a help file, followed by a list of mnemonics corresponding to all possible error types. • Ba[d Blocks]. Outputs a report of possible bad blocks detected from the disk errors logged in the input file entered in Step 2. The next dialogue step is 11. • Su[mmary]. Outputs a report of the number of errors detected for each error type and for each unit number within a specific error type. The next dialogue step is 10. • Fu[ll]. Outputs a report of the detailed contents of each selected error record in the error file. Selection of a specific record depends on the answers to steps 5 through 9 in the ERRDIS dialogue. • Ze[ro]. Zeroes the error file, either completely or by error type. The next dialogue step is 12.
5	Specific Error Type <All>? Enter a two-character mnemonic identifying the specific error type that you want ERRDIS to process. The default answer selects all error types. Note that if you select All, you may also attach the /NOTAPE switch if you want magnetic tape errors to be omitted from the error listing.
6	Starting Date <First Error>? Enter the date of the earliest error that you want ERRDIS to process. The format is dd-mmm-yy or yy.mm.dd. The default answer is the date of the first error in the error log file. If you press Return to select the default, the next step is 8.
7	Starting Time <First Error>? Enter the time of the earliest error that you want ERRDIS to process. The format is hh:mm, either in 24-hour time or with AM/PM afterwards. The default is the time of the first error in the error log file on the selected date.
8	Ending Date <Last Error>? Enter the date of the latest error that you want ERRDIS to process. The format is dd-mmm-yy or yy.mm.dd. The default is the date of the last error in the error log file. If you select the default, then the next dialogue step is 11.
9	Ending Time <Last Error>? Enter the time of the latest error that you want ERRDIS to process. The format is hh:mm, either in 24-hour time or with AM/PM afterwards. The default is the time of the last error in the error log file on the selected ending date. The next dialogue step is 11.

(continued on next page)

Table 10–1 (Cont.): ERRDIS Dialogue Explanation

Step	Prompt and Description
10	List Bad Blocks (Yes/No) <No>? Press Return (or type No) if you do not want ERRDIS to generate a list of possible bad blocks following the summary report. Type Yes if you want a list of bad blocks.
11	Zero Error File upon completion (Yes/No) <No>? Type Yes to have ERRDIS zero the previously specified error log file after it successfully generates the desired report. (You must type all three characters of Yes to zero the file.)
12	Zero Which Error Type <All>? Press Return or type All to have ERRDIS zero all error types in the file. Otherwise, enter the error type or types you want zeroed.

After you finish the dialogue, ERRDIS:

- Processes the selected error records
- Formats and generates the selected report
- Zeros the error log file (if requested in step 11 of Table 10–1)
- Returns to step 2 of Table 10–1.

At that point, you can enter Ctrl/Z in response to any question to return the terminal to the keyboard monitor prompt.

10.2.2 Help Report

The Help report prints information on the use of the ERRDIS program and then prints a list of the error mnemonics other programs use. After you run ERRDIS and select the Help report, the program prints the Help text.

10.2.3 Summary Report

A summary report at the end of this section supplies general information on all errors logged. The first two columns list the two-character error mnemonic and full error description. The TOTAL REC/LOG column lists the total errors received by the ERRCPY program and the total it stored in the error file. These two totals may differ due to the limits on:

- The number of errors that may be logged
- The size of the error file

The UNIT NUMBERS column lists the total number of errors generated by each device on a controller.

The CONTROLLER ERRORS column indicates how many of the errors logged under TOTAL REC/LOG are controller errors, as opposed to unit errors. (Controller errors apply to the controller itself, while unit errors apply to a specific unit being controlled.) If N/A appears under CONTROLLER ERRORS, that means that ERRDIS is not keeping track of CONTROLLER ERRORS for that particular ERROR CODE-DESCRIPTION.

The asterisk (*) next to 239/100 indicates that MM, under the ERROR CODE-DESCRIPTION column, has exceeded the maximum number of errors (100) that the Error Package logs for that particular error type.

After you run ERRDIS and select the SUMMARY report, the program prints the summary as follows.

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7   EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
He[lp], Ba[d Blocks], Su[mmary], Fu[ll] Report or Ze[ro] File <Summary>? <RET>
List Bad Blocks (Yes/No) <No>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Summary Report taken on 29-May-89, 10:50 AM
     Input File: ERROR$:ERRLOG.FIL  Output File: KB:ERRDIS.OUT
     Reported Date/Time Range:
        13-May-89, 12:21:22 AM through 28-May-89, 12:35:47 AM

      ERROR            TOTAL          UNIT NUMBERS        CONTROLLER
CODE-DESCRIPTION      REC/LOG   0  1  2  3  4  5  6  7      ERRORS

PF Power Fail/Start-up  1/1                                  N/A
DU MSCP Disks           7/7        1        2     1          3
DB RH11/RP04-05-06      3/3           3                      N/A
DL RL01/RL02            3/3     1     2                      N/A
MM RH11/TU16/TE16   * 239/100   49 7     5  2     37         N/A
SH MSG from SHUTUP      1/1                                  N/A

Total of 115 Errors Logged out of 254 Received
34 out of 100 Blocks have been used in ERROR$:ERRLOG.FIL

********************************************************************
Input File <ERROR$:ERRLOG.FIL>? <Ctrl/Z>

$
```

10.2.4 Bad Block Report

The ERRDET program reports possible bad blocks that were detected by the monitor by checking the error logging file for such entries. The criteria for a possible bad block depends on the type of disk.

Each bad block report includes:

- The logical block number
- The logical name or pack identification of the disk
- The physical device name of the disk

Using the logical block number that ERRDIS prints, you can add the possible bad block to the bad block file with the BADS suboption of REFRESH. See the *RSTS/E System Installation and Update Guide* for information on adding bad blocks.

The following example shows a list of bad blocks that the ERRDIS program prints:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7   EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? BAD <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Bad Block Report taken on 29-May-89, 10:51 AM
     Input File: ERROR$:ERRLOG.FIL  Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        First Error through Last Error
     ERROR$:ERRLOG.FIL will not be zeroed upon completion

List of Possible Bad Blocks:
----------------------------
Logical Name            RAIDER
Physical Name           DM1:
Logical Block Number    3401

Logical Name            OFTHE
Physical Name           DR4:
Logical Block Number    237968

Logical Name            LOST
Physical Name           DR5:
Logical Block Number    432952

Logical Name            ARK
Physical Name           DM2:
Logical Block Number    8

Total Number of Possible Bad Blocks: 4

********************************************************************
Input File <ERROR$:ERRLOG.FIL>? ^Z
```

NOTE

Use caution before adding possible bad blocks (that ERRDIS reports) to the bad block file. Instead of corrupt disk surfaces, hardware controller or disk drive problems may have caused the errors to occur.

10.2.5 Adding Bad Blocks to the Bad Block File

The ERRDIS program prints a list of potential bad blocks. If you decide to add a block to the bad block file BADB.SYS on a disk, use the BADS suboption of the REFRESH option in INIT.SYS (see the *RSTS/E System Installation and Update Guide*). In preparation for adding bad blocks, which requires you to bring the system down, use the PIP system program to copy the data in a bad file to a new file. As your system discovers bad blocks, PIP lets you collect this information in a single file. You can then use this file to add the bad blocks to BADB.SYS when you normally take your system down.

Follow these steps to use PIP for this purpose:

1. Specify the /GO switch (along with other necessary switches, such as /CL, and so forth) to ignore possible ?Data error on device errors. Rename the file that contains the bad block, but do not delete it:

   ```
   $ RUN $PIP
   *FILE.TMP=FILE.OLD/GO
   *FILE.BAD=FILE.OLD/RE
   *FILE.OLD=FILE.TMP/R
   ```

2. When it is convenient, shut down the system and use the BADS suboption of REFRESH to add any bad blocks to the disk's bad block file. Each new bad block is now allocated to two files ([0,1]BADB.SYS and FILE.BAD in the previous example).
3. The BADS suboption automatically rebuilds the allocation tables. It then tell: you the file has a bad block and allows you to delete the file (FILE.BAD in the previous example). The *RSTS/E Installation and Update Guide* describe: this operation.

When BADS deletes the file, it frees the blocks in the file that are not bad.

You must modify the previous procedure if the bad block is in a file directory. In this case, use the BACKUP program to copy all accessible files in the account (or disk) to an archive medium. When you add the bad block and clean the disk, you can delete the account whose directory contains the bad block. After starting timesharing, use BACKUP to restore the files from the archive medium.

The bad block report does not apply to MSCP disks; bad blocks on those devices are recovered automatically.

10.2.6 Full Report

Each error displayed includes:

- A title line containing the error code mnemonic
- The error description
- A sequence number
- The date and time the error occurred

The monitor maintains the sequence numbers, starting at one, from start-up to shutdown. A crash and subsequent automatic restart does not reset these numbers.

There are four basic categories for individual errors displayed within the full error report:

- Nonperipheral errors such as processor traps and memory parity
- Disk errors
- Nondisk peripheral device errors such as magnetic tape errors
- Other errors such as the shutdown message received from SHUTUP or missed errors

The program automatically checks for and reports possible bad blocks for disk errors included in the scan. The following sections contain partial listings of a FULL report.

When the monitor detects a hardware or software error, it sends information about the error to the ERRCPY program. ERRCPY can accept and file error information until the number of queued errors reaches 40. The monitor logs all errors it detects beyond this point as "missed errors." Only after the queue begins to empty and the number of errors queued to ERRCPY falls below the maximum of 40 can the monitor begin to send the normal error information. This means that while the queue is full, the monitor reports all subsequent errors as "missed errors," rather than sending the usual error information to ERRCPY.

The monitor also reports missed errors if it does not have a sufficient number of general small buffers available to store error information. When the number of general small buffers drops below 75, the monitor reports any detected error to ERRCPY as a "missed error."

10.2.6.1 User Description in Full Report

All errors except shutdown, missed errors, and some MSCP controller errors usually include a user description. The name fields in the user description relate to a specific job whose number appears in the first field (Job number). (Under certain conditions, a job number of 0 is possible. In this case, the user description is not listed.)

The following example shows a complete error report for a tape device; the report includes both a user description and a detailed description:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7  EDERE::
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? <RET>
Starting Date <First Error>? 13-May-89 <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (All Types) taken on 29-May-89, 02:17 PM
     Input File: ERROR$:ERRLOG.FIL     Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        13-May-89 (First Error) through Last Error
     ERROR$:ERRLOG.FIL will not be Zeroed upon completion

********************************************************************
MM  RH11/TU16/TE16 Seq #92 Occurred on 28-May-89 at 02:11:44 PM

User Description:
-----------------
Job Number               18
KB Number                2
Account                  [1,170]
Program Name             PASIKO
User Job Physical Addr.  02310000
User Job Size            28K
Control Parameters       200
RTS Name                 RT11
RTS Physical Address     01714000

Detailed Description:
---------------------
Timeout Indicator        000
Physical Name            MM0:

DDB                      017600  173444  104252  000011
                         045471  005171  001630  177400
                         000000  147240  000000  001007
                         000056  002004  127311

CSR Address:             176700
------------
```

```
Contents of Registers:
----------------------
MTCS1       145270          TRE Xfer Err
MTWC        000400
MTBA        132444
MTFC        000000
MTCS2       000100
MTDS        150660          PE Mode
MTER        100000          COR/CRC
MTAS        000001
MTCK        000000
MTMR        000000
MTDT        000000          TU77 TM03
MTSN        000000
MTTC        000000
MTBAE       000016
MTCS3       002000
```

Table 10–2 summarizes the user description data.

Table 10–2: User Description Data

Heading	Meaning
Job Number	For disk errors, the number of the job that requested the I/O; for nondisk peripheral errors, the job owning the device; for nonperipheral errors, the job running at the time of the error.
KB Number	The keyboard number of the job, followed by DET if the job is detached.
Account	The PPN of the user who created the job.
Program Name	The name of the program running in the job virtual address space (low segment).
User Job Physical Address	The current physical memory address of the job.
User Job Size	The size of the user program running in the low segment of the job virtual address space.
Control Parameters [1]	Monitor control information.
RTS Name	The name of the run-time system mapped into the highest segment of the user's virtual address space.
RTS Physical Address	The address of the location in physical memory where the run-time system resides.

[1] A detailed description of this entry is beyond the scope of this manual. The entry is intended for use by DIGITAL personnel.

10.2.6.2 Disk Error Detailed Description

A FULL report for a disk error includes three sections:

- User description
- Detailed description
- Contents of registers

Table 10–3 describes the information found in the detailed description portion of a disk error report. Before you refer to the table, study the following example.

The example shows what a single complete disk error looks like. If the entire Full Report of DB type errors were reproduced here, you would see two more identically structured error messages following this one. As you can tell from the sample in the section "Summary Report," there are three DB errors under TOTAL REC/LOG. The example follows:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7  EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? DB <RET>
Starting Date <First Error>? 13-May-89 <RET>
Starting Time <First Error>? <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (DB only) taken on 29-May-89, 11:00 AM
     Input File: ERROR$:ERRLOG.FIL      Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        13-May-89 (First Error) through Last Error
     ERROR$:ERRLOG.FIL will not be Zeroed upon completion

********************************************************************
DB  RH11/RP04-05-06 Seq #81 Occurred on 28-May-89 at 07:12:14 AM

User Description:
-----------------
Job Number               17
KB Number                33 (Det)
Account                  [1,100]
Program Name             LOGOUT
User Job Physical Addr.  02744000
User Job Size            11K
Control Parameters       200
RTS Name                 ...RSX
RTS Physical Address     00000000

Detailed Description:
---------------------
I/O Status               000
Timeout Indicator        000
Offset Position          Previous
Overlapped Seek Ind.     377
Unit Size in DC's        121510
Device Cluster Size      8
Pack Cluster Size        8
Logical Name             D
Physical Name            DB2:
Logical Block Number     19272

DSQ                      004500  173442  104252  000011
                         045501  006171  064540  174000
                         000000  146300  000000  001000
                         000056  002004  127310  001003

WCB                      100000  004442  000002  000021
                         146274  000000  000000  001310
                         113570  004550  004551  005165
                         005166  005413  005414  005415

FCB                      147040  070560  000544  076725  [1,100]
                         064511  077430  037404  000001  TEMP33.TMP
                         001300  113570  001270  113570
                         000011  000226  000010  146300

CSR Address:             176700
------------
```

```
Contents of Registers:
----------------------
RPCS1     145270        TRE Xfer Err
RPWC      000400
RPBA      000000
RPDA      000000
RPCS2     000000
RPDS      000000        Not Ready
RPER1     000000
RPAS      000000
RPLA      000000
RPMR      000000
RPDT      000000
RPSN      000000
RPOF      000000
RPDC      000000
RPCC      000000
RPER2     000000
RPER3     000000
RPEC1     000000
RPEC2     000000
```

Table 10–3 describes the information found in the detailed description portion of the previous disk error report. The program prints an additional field if the error signalled a possible bad block.

Table 10–3: Disk Error Detailed Description

Heading	Meaning
I/O Status	See following Timeout Indicator.
Timeout Indicator	Timeout and I/O Status are considered as a pair. One of the following states can exist: Timeout Indicator — I/O Status — Meaning 0 — 0 — Idle x — 1 — SEEK in progress 0 — 1 — Waiting for Read/Write 1 — x — Timeout where x equals any nonzero value.
Offset Position	The contents of Offset Position for disks having offset capabilities: 0 = centerline position 20 = +x offset 220 = -x offset 40 = +2x offset 240 = -2x offset 60 = +3x offset 260 = -3x offset 100 = offset unknown where x equals the number of micro inches of offset (a drive dependent parameter).
Overlapped Seek Ind.	377 = overlap seek driver 000 = non-overlap seek driver You determine which of these two values appears during system installation.
Unit Size in DC's	Size (in octal) of the disk expressed as the number of device clusters.

(continued on next page)

Table 10-3 (Cont.): Disk Error Detailed Description

Heading	Meaning
Device Cluster Size	Device cluster size for this drive.
Pack Cluster Size	Pack cluster size for this disk.
Logical Name	The pack identification if no system wide logical name was given at mount time or the system logical name entered at mount time.
Physical Name	The physical name and unit number.
Logical Block Number	The block number where the error occurred.
DSQ [1]	The disk request queue entry block.
WCB [1]	The WCB (Window Control Block) is present for disk errors that resulted from a file request by the user job.
FCB [1]	The File Control Block.
CSR Address	The CSR address is the base address of the set of registers in the I/O page belonging to the device that caused the error.
Contents of Registers	This section contains the name of each readable device register, its contents and, possibly, an abbreviated description of various error bits that were found to be set to one in the register. ERRDIS stores a maximum of 21 error bits for each error type. The data file ERRDAT.FIL contains the register mnemonics, error bit descriptions, and error code descriptions for all error types. ERRDIS requires this file to run. See the *PDP-11 Peripherals Handbook* for a detailed description of registers and bit descriptions. The register mnemonics displayed by ERRDIS correspond to those given in the manual.

[1] A detailed description of this entry is beyond the scope of this manual. The entry is intended for use by DIGITAL maintenance personnel.

10.2.6.3 MSCP Variations on the Full Report

For systems that support MSCP type disk hardware, (for example, the RA80), there are two variations on the Full Report:

- If there is a disk error, then the Full Report looks just like the normal Full Report, except that it has two extra sections called the MSCP Description and the Driver Related Data (see Example #1). The Full Report includes the following five sections:
  - User Description
  - Detailed Description
  - Contents of Registers
  - MSCP Description
  - Driver-related Data
- If there is a controller error, then the Full Report shows an abbreviated error message. The Full Report does not have a User Description, and its Detailed Description is not as large as in the normal Full Report (see Example #2). The Full Report includes the following four sections:
  - Detailed Description
  - Contents of Registers

- MSCP Description
- Driver-related Data

If you refer back to the sample report in the section "Summary Report," you will notice that the first example that follows is the error that was logged under UNIT NUMBER 1. The second sample error message that follows is one of the three errors logged under CONTROLLER ERRORS.

Example #1:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7 EDERE
Input File <0,3]ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? DU <RET>
Starting Date <First Error>? <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (DU only) taken on 29-May-89, 02:17 PM
     Input File: ERROR$:ERRLOG.FIL      Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        First Error through Last Error
     ERROR$:ERRLOG.FIL will not be Zeroed upon completion
********************************************************************
DU  MSCP Disks Seq  46 Occurred on 25-May-89 at 07:04:02 PM

User Description:
-----------------
Job Number                   12
KB Number                    16
Account                      [1,217]
Program Name                 PACKID
User Job Physical Addr.      01610000
User Job Size                7K
Control Parameters           000
RTS Name                     BAS4F
RTS Physical Address         01514000

Detailed Description:
---------------------
I/O Status                   000
Timeout Indicator            377
Offset Position              000
Overlapped Seek Ind.         000
Unit Size in DC's            000000
Device Cluster Size          0
Pack Cluster Size            0
Logical Name
Physical Name                DU7:
Logical Block Number         200806

DSQ                          000000  177430  103664 001424
                             010147  000367  023000 000400
                             000010  000000  000000 075576
                             000003  004000  000000 003777


CSR Address:                 160414
------------

Contents of Register:
---------------------
SA        000000             Controller On Line

MSCP Description:
-----------------
Packet Type                  End Message
```

```
MSCP Envelope                000060  000001

MSCP Packet                  074130  140416  000007  000000
                             000203  000043  000736  000000
                             000000  000000  001624  000000
                             020424  001003  031431  020144
                             000007  000000  000037  000001
                             000001  003400  000020  002001

Command Reference Number     074130 140416
Drive Number                 000007
Message Flags                000
End Flags                    000
Endcode                      203       GET UNIT STATUS command
Status/Event Code            000043    No vol mntd or r/s switch out
First Bad Block              543437593.

Driver Related Data:
--------------------
Internal Control Status Word 000300
Internal Unit Status Word    000000
Driver Error Code            000007  GUS failed; unit not functional
BBR Flag Word                000000
```

Example #2:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7  EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? DU <RET>
Starting Date <First Error>? <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (DU only) taken on 29-May-89, 02:19 PM
     Input File: ERROR$:ERRLOG.FIL      Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        First Error through Last Error
     ERROR$:ERRLOG.FIL will not be Zeroed upon completion

********************************************************************
DU  MSCP Disks Seq  2 Occurred on 21-May-89 at 12:59:04 PM

Detailed Description:
---------------------
I/O Status                   000
Timeout Indicator            377

CSR Address:                 172150
------------

Contents of Register:
---------------------
SA        000000             Controller On Line

MSCP Description:
-----------------
MSCP Packet Type             Last Fail Packet

MSCP Envelope                000030  000020

MSCP Packet                  000000  000000  000000  000000
                             000400  000012  000000  000000
                             000000  000406  000004  000013


Controller Class             001      Mass Storage
Controller Model             006      UDA50-A
Controller Hardware Version  000
Controller Software Version  004
UDA Internal Error Code      000013   ER.MST - Bus Master Error
```

```
Driver Related Data:
--------------------
Internal Control Status Word 000102
Internal Unit Status Word    000000
Driver Error Code            000000    Driver did not detect error
BBR Flag Word                000000
*********************************************************************
```

10.2.6.4 Nondisk Peripheral Device Error Detailed Description

Table 10–4 lists the fields found in the detailed description portion of the nondisk peripheral device error report. Before you refer to the table, study the following example. The example shows a single complete tape error. As you can tell from the example in the section "Summary Report," this is one of the 239 tape errors that were received. This particular error was one of the 100 logged, and also one of the seven logged under UNIT NUMBER 1. The entire Full Report contains 99 more tape error messages. The example follows:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7 EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? MM <RET>
Starting Date <First Error>? <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (MM only) taken on 29-May-89, 02:17 PM
     Input File: ERROR$:ERRLOG.FIL      Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        First Error through Last Error
     ERROR$:ERRLOG.FIL will not be zeroed upon completion

*********************************************************************
MM  RH11/TU16/TE16 Seq #44 Occurred on 28-May-89 at 05:51:08 AM

User Description:
-----------------
Job Number                 11
KB Number                  33 (Det)
Account                    [1,100]
Program Name               BACKUP
User Job Physical Addr.    02610000
User Job Size              15K
Control Parameters         200
RTS Name                   ...RSX
RTS Physical Address       00000000

Detailed Description:
---------------------
Timeout Indicator          000000
Physical Name              MM1:

DDB                        000016  000426  002102  100001
                           000001  000000  000000  000000
                           105536  120540  010004  000021
                           000361  001002  010760  000053
                           000000  007314  004400

CSR Address:               172440
------------
```

```
Contents of Registers:
----------------------
MTCS1     145260                TRE Xfer Err
MTWC      000000
MTBA      010004
MTFC      000000
MTCS2     000100
MTDS      150760                PE Mode
MTER      100000                COR/CRC
MTAS      000001
MTCK      000010
MTMR      001100
MTDT      142011                TU16/TE16
MTSN      060023
MTTC      102301
```

Table 10–4 describes the information found in the detailed description portion of the previous nondisk peripheral device error report example.

Table 10–4: Nondisk Peripheral Device Format

Heading	Meaning
Timeout Indicator	Nonzero indicates that an expected response was not received from the device in the allotted time.
Physical Name	The physical name and unit number of the device.
DDB[1]	The DDB (Device Data Block) contains various parameters required by the system to control the device.
DDB Extension[1]	Supplementary DDB information.
CSR Address	The CSR address is the base address of the set of registers in the I/O page belonging to the device that caused the error.
Contents of Registers	This section contains the name of each readable device register, its contents and, possibly, an abbreviated description of various error bits that were found to be set to one in the register. ERRDIS stores a maximum of 21 error bits for each error type. The data file ERRDAT.FIL contains the register mnemonics, error bit descriptions, and error code descriptions for all error types. ERRDIS requires this file to run. See the *PDP-11 Peripherals Handbook* for a detailed description of registers and bit descriptions. The register mnemonics displayed by ERRDIS correspond to those given in the manual.

[1] A detailed description of this entry is beyond the scope of this manual. The entry is intended for use by DIGITAL maintenance personnel.

10.2.6.5 Nonperipheral Error Detailed Description

See the related processor handbook for detailed descriptions of nonperipheral error fields. Table 10–5 gives an abbreviated description of these items. Before you refer to the table, study the following example. This example shows a single complete error message extracted from the ERRLOG.FIL using ERRDIS to reproduce only the PF type errors:

```
$ RUN ERROR$:ERRDIS <RET>
ERRDIS  V9.7  RSTS V9.7 EDERE
Input File <ERROR$:ERRLOG.FIL>? <RET>
Output to <KB:ERRDIS.OUT>? <RET>
Help, Bad Blocks, Summary, Full Report or Ze[ro] File <Summary>? FULL <RET>
Specific Error Type <All>? PF <RET>
Starting Date <First Error>? <RET>
Ending Date <Last Error>? <RET>
Zero Error File upon completion (Yes/No) <No>? <RET>
  ERRDIS Full Report (PF only) taken on 29-May-89, 02:21 PM
     Input File: ERROR$:ERRLOG.FIL      Output File: KB:ERRDIS.OUT
     Requested Date/Time Range:
        First Error through Last Error
     ERROR$:ERRLOG.FIL will not be zeroed upon completion

********************************************************************
PF  Power Fail/Start-up Seq #1 Occurred on 13-May-89 at 02:15:01 PM

Detailed Description:
---------------------
R0                       000000
R1                       004200
R2                       000000
R3                       000002
R4                       004100
R5                       000000
Virtual PC               000077
Physical PC            00000077
Processor Status         000000
Stack Pointer            002074
(SP)                     000002
(SP+2)                   000000
(PC-6)                   034156
(PC-4)                   004356
(PC-2)                   034156
(PC)                     004357
CPU ID                       -1
CPU ERR                  000000
```

Table 10–5 describes the information found in the detailed description portion of the previous example.

Table 10–5: Nonperipheral Error Format

Heading	Meaning
R0-R5	Processor registers 0 through 5. Each register is listed on a separate line.
Virtual PC	The address (within the user virtual address space) of the instruction being executed at the time of the error.
Physical PC	The address (in the physical memory address space) of the instruction being executed at the time of the error.
Processor Status	The processor status word.
Stack Pointer	The address of the current top of stack.
(SP) (SP+2)	The contents of the top two words in the stack.
(PC-6) (PC-4) (PC-2) (PC)	The contents of the three words preceding the current program PC and the contents of the current PC.
CPU ID	If it exists, the contents of the CPU identification register.
CPU ERR	If it exists, the contents of the CPU error register.
MED X	If it exists, the CPU has Maintenance Examine/Deposit Instruction.
Contents of Registers	Certain nonperipheral errors include contents of key registers. The format is identical to the contents of registers for disk and nondisk peripheral errors.

NOTE

The PF error code is used for recording both start-up and power fail recovery. The error log for a normal start-up has a sequence number of 1. For a power fail restart, the sequence number is greater than 1.

10.2.7 Zero File

The Zero File option of ERRDIS lets you clear any or all types of errors from the ERRLOG.FIL file without generating a report. After you select this option, ERRDIS prompts you for the error type to zero. The default is to zero *all* error types. Enter the two-character mnemonic for the error type you wish to clear, or type Return to accept the default. The following example shows how to clear all DB errors:

```
$ RUN ERROR$:ERRDIS
ERRDIS  V9.0  RSTS V9.7 EDERE
Input File <ERROR$:ERRLOG.FIL>?
Output to <KB:ERRDIS.OUT>?
He[lp], Ba[d Blocks], Su[mmary], Fu[ll] Report or Ze[ro] File <Summary>? ZERO
Zero Which Error Type <All>? DB

Input File <ERROR$:ERRLOG.FIL>?
```

10.3 Analyzing System Crashes: ANALYS

Time-sharing operations halt when a crash occurs on a RSTS/E system. The system then writes an image of read/write memory, tables, and XBUF to the CRASH.SYS file in account [0,1]. The system bootstraps the system disk, loads the initialization code into memory, and executes an automatic restart if the CPU switch register has bit 0 set.

Unless you run the ANALYS program to save crash dump information, the next system crash causes the CRASH.SYS file to be overwritten and the information in the file to be lost. Normally, this is done as part of system start-up. See Chapter 2 for a description of automatic system recovery procedures.

The crash analysis program ANALYS consists of four modules:

- ANALYS
- ANALY1
- ANALY2
- ANALY3

For simplicity, the documentation refers to only one program, ANALYS, which functionally includes the four modules.

10.3.1 Running the ANALYS Program

Run the ANALYS system program by typing the command:

```
$ RUN ERROR$:ANALYS<RET>
```

After you press Return, the ANALYS program prints an identification line and three dialogue questions:

```
ANALYS V9.7  RSTS V9.7  EDERE
INPUT <[0,1]CRASH.SYS>? <RET>
OUTPUT <ANALYS.DMP<188>>? <RET>
Crash error log filename<ERROR$:ERRCRS.FIL>? <RET>

$
```

The ANALYS program uses the following switches:

- /DET tells ANALYS to detach.
- /[NO]DUMP tells ANALYS whether or not to include a memory dump in its output. The default is /DUMP.
- /NARROW tells ANALYS whether to use 80 or 132 columns in its output. The default is 132.
- /NOSTB tells ANALYS not to include the symbol table in its output.
- /NOXBUFDUMP tells ANALYS not to include the octal dump of XBUF. It also implies a /DUMP, so you do not need to include /DUMP with it.
- /SIL lets Digital analyze the crash and .SIL files submitted with a Software Performance Report (SPR).

Table 10–6 explains how to answer the ANALYS program questions.

Table 10–6: ANALYS Program Dialogue

Question and Response

INPUT <[0,1]CRASH.SYS>?

Asks for the name of the file to be analyzed. Type the file specification of the file you want ANALYS to analyze, which by default is CRASH.SYS in account [0,1]. Press Return to accept the default response. However, you may want to keep copies of different crash files.

ANALYS uses the currently installed monitor SIL file to extract symbolic references. If the crash file being analyzed is not associated with the currently installed monitor, you must append the /SIL: switch with the name of the related monitor SIL. This switch allows DIGITAL to analyze crash and .SIL files submitted with a Software Performance Report (SPR).

OUTPUT <ANALYS.DMP<188>>?

Requests a disk file or a device designator for the output medium. You can select the default output file ANALYS.DMP by pressing Return. Enter another file specification if you do not want ANALYS to place the output in the default account. ANALYS always prints an annotated version of a memory dump and automatically prints a memory dump in 132-column format unless you:

1. Request output to your terminal (for example, KB:ANALYS.DMP).
2. Append the /NODUMP switch to omit the memory dump.
3. Append the /NOXBUFDUMP switch to omit the octal dump of XBUF.
4. Append the /NARROW switch to a file specification (or type /NARROW and press Return in response to the OUTPUT question if you want ANALYS.DMP to be the output to be the output file) for 80-column output.
5. Append the /NOSTB switch to omit the symbol table. Do not use this switch when submitting an SPR.

In any case, you get the annotated output in 80-column format, but with fewer memory locations printed on each line. If you do not want a dump of memory, append the /NODUMP switch to your response.

Crash error log filename<[0,3]ERRCRS.FIL>?

Asks for the file specification of the file you want to hold certain error information from the crash file. Press Return to have ANALYS write this information to the ERRCRS.FIL in the error package account [0,3].

Note that the /DET qualifier causes ANALYS to detach.

ANALYS retrieves error information saved at the time of a system crash but not written to the system error logging file ERRLOG.FIL. The ERRDIS program accepts the file you specify as input and produces an error log report from its contents. To retain a single continuous error logging file, use the ERRINT option.

Normally, ANALYS takes more than 30 minutes to run. When the output stops, the program automatically terminates and returns to the keyboard monitor prompt.

10.3.2 ANALYS Output

Output of the ANALYS system program supplies valuable hardware and software information that a software specialist can use to determine possible causes of system crashes. It includes:

- A report similar to SYSTAT
- A memory dump of the critical contents of memory
- A listing of all monitor symbols

The report also contains DECnet/E information if you configured your system with DECnet/E. See the *RSTS/E Utilities Reference Manual* for an explanation oi SYSTAT.

Table 10–7 shows the error codes that ANALYS reports in the crash dump data.

Table 10–7: System Crash Error Code

Error Code (Octal)	Meaning
-1(177776)	Power fail error.
-2(177776)	Jump to 0.
-3(177775)	This code is returned when RSTS/E does a crash dump as a result of halting the CPU and continuing from location 52 (octal).
-4(177774)	This code is returned when RSTS/E detects an unrecoverable problem and forces a software crash. This can be caused by either a RSTS/E software problem or faulty harware.
41	Trap to 4.
42	Trap to 10.
43	Trap to 250 (Memory management violation).
44	Kernel SP Stack overflow.
46	Trap to 114 (Parity memory error).

To obtain crash information printouts automatically, include the proper commands in the START.COM file. The commands run ANALYS to preserve the crash information and run ERRDIS to create a report. See Chapter 3 for a description of the commands you should include in the START.COM file.

Chapter 11

Device Management and Caching Control

This chapter introduces the concepts of device management and caching control on your RSTS/E system. Later, it explains DCL commands available to you in more detail. Table 11–1 summarizes these DCL commands.

Table 11–1: DCL Commands for Device Management and Caching Control

Command	Meaning
	Device Management
ALLOCATE	Allocates a device and optionally assigns it a logical name
DEALLOCATE	Deallocates the specified device
SET DEVICE	Sets the characteristics for the specified device
SET PRINTER	Set the characteristics of the specified line printer
SHOW DEVICE	Displays information about the specified device
SHOW DISKS	Displays the status of all mounted disks
SHOW PRINTER	Displays the characteristics of the specified printer
	Caching Control
SET CACHE	Enables data and directory caching
SHOW CACHE	Displays the current cache settings
CREATE/VITUAL_DISK	Reserves a section of memory for use as a virtual disk
DELETE/VIRTUAL_DISK	Frees the seciton of memory reserved for the virtual disk

11.1 Principles of Device Management

Use the SET DEVICE command to set device attributes. You need HWCTL, HWCFG, and DEVICE privileges to use the SET DEVICE command.

You can disable a specific device for the current time-sharing session with the /DISABLE qualifier to the SET DEVICE command. To disable a device for subsequent sessions, use the SET DEVICE command in the START.COM file. To reenable a disabled device, use the /ENABLE qualifier.

The /RESTRICT qualifier limits the use of the specified device to users with DEVICE privilege. The /NORESTRICT qualifier lets any user allocate or open the specified device.

11.2 Optimizing Disk Directory Structure: REORDR

The REORDR program can restructure the disk directories on your system to improve disk access time. This process begins once you run REORDR and answer the set of questions in its dialogue. The following sections discuss:

- Operations that REORDR performs
- REORDR dialogue
- Restructuring the public disk structure
- Example of the entire process

11.2.1 Why Use REORDR?

The system catalogs RSTS/E files on disk in user file directories (UFD). As you create, delete, and extend files, the file directory entries become scattered across the surface of the disk. This scattering increases both the time to open files and the time to retrieve data blocks.

The REORDR program can perform three operations to restructure disk directories. Each of these operations can improve the performance of your system. After you run the program and answer a sequence of dialogue questions, REORDR performs the following:

1. Places the list of file names in as few physical disk blocks as possible. This reduces the number of directory accesses the system needs to make for a file open operation.
2. Attempts to place all directory information concerning the physical location of file data (retrieval pointers) for a file into contiguous disk blocks. This reduces the number of disk directory accesses required to perform disk read and write operations.
3. Optionally sorts the file name list in one of four ways, depending on how you answer the dialogue questions. This reduces the number of directory accesses the system needs to make for a file open operation.

When RSTS/E creates a new file in an account, it adds the file name at the end of the current directory list. That is, RSTS/E places files last in the directory if, during disk initialization, you typed NO to the DSKINT question NEW FILES FIRST (see the *RSTS/E System Installation and Update Guide*). Answering "Yes," while not efficient under most situations, may be more efficient for your installation. You should have new files first only if users on your system do not create many new files and their accounts are not large. Otherwise, because of the way RSTS/E retrieves user directory information, it is more efficient to have new files last. With the REORDR program, you can choose to organize files by access date or creation date.

Note that DCL opens all work files in DCLWRK$:, or in your own account if DCLWRK$: does not exist. In either case, the account with this file in it cannot be reordered.

11.2.2 Dialogue Questions and Responses

The REORDR program prints a set of questions after you type RUN AUXLIB$:REORDR:

```
$ RUN AUXLIB$:REORDR
REORDR V9.0 RSTS V9.0 EDERE

Sort Directory(s) (YES/NO) <NO>? YES

Order by CRE[ATION] or ACC[ESS] Date<CRE>? <RET>

In FOR[WARD] or REV[ERSE] Order<FOR>? <RET>

Device and UFD Specification(s)? DB1:[*,*],DB2:[*,*]
```

Table 11–2 summarizes these questions and their responses.

Table 11–2: REORDR Dialogue Questions

Question and Response
Sort Directory(s) (YES/NO) <NO>?
Type NO or press Return to keep the existing order of the account(s) you ar processing. Either response causes REORDR to skip the next two questions Type YES to have REORDR sort the account(s) you designate. REORDR asks the next two questions to determine how you want the account(s) sorted.
Order by CRE[ATION] OR ACC[ESS] Date<CRE>?
Type CRE or press Return to have REORDR sort the account(s) by creation date. Type ACC to have REORDR use ACCESS date to sort the account(s). Depending on how the disk was initialized, the access date can have one of two meanings. It can mean: • When the file was last accessed • When it was last modified Note that the abbreviation DLW in the comments part of the SHOW DISKS command or the DISPLY program shows that a disk's access date is set to the date of last modification.
In FOR[WARD] or REV[ERSE] Order<FOR>?
Type FOR or press Return if you want REORDER to sort oldest files first. Type REV to have REORDR create a sorted file with newest files first.

(continued on next page)

Table 11-2 (Cont.): REORDR Dialogue Questions

Question and Response

Device and UFD Specification(s)?

Enter the device mnemonic(s) and unit number(s) of the device(s) you want REORDR to process. Include the account(s) you are reordering with the device designator. For example, the specification DB0:[1,210],DB1:[*,210] is a valid response. Although SY: (indicating the entire public structure) is not a valid device, SY0: (indicating the system disk, the device that was bootstrapped) is acceptable for the device response. An asterisk (*) is a valid character in either or both of the project-programmer account fields. It tells REORDR to process all project or all programmer numbers on the device. You can also include multiple specifications if you separate them with commas (,). The account designators are:

- [*,*]
  Process all user file directories on the disk
- [p,*]
  Process all user file directories with project number p
- [*,pn]
  Process all user file directories with programmer number pn
- [p,pn]
  Process only the user file directory with project number p and programmer number pn

NOTE

For disks initialized prior to V8.0, REORDR can not order account [1,1], even though it may be implicit in an account designation.

11.2.3 Reordering Your Disks

To reorder all disks in the public disk structure, you must include the device designator for each disk and the account numbers of the files you want reordered:

```
Device and UFD Specification(s)? DB0:[*,*], DB1:[*,*]
```

REORDR orders all the directories in the UFD on RP06 unit 0 and then all the directories in the UFD on RP06 unit 1. Make sure you logically mount the disks with write access before running the program.

To prevent damage to a directory, REORDR requires that all files in a UFD be closed during the entire operation.

NOTE

REORDR verifies that no file is open when it starts to process the UFD. However, it is unable to detect file opening, creation, or deletion between the initial check and the completion of the UFD processing. REORDR can damage the file structure of a disk if a file creation takes place during reordering. For this reason, follow either of these steps as a precaution:

- Disable logins with the SET SYSTEM/NOLOGINS command. Make sure that no other jobs (including spoolers and detached jobs) are active on the system.
- Dismount the disk to be reordered and remount it using the /NOSHARE qualifier.

The only safe way to reorder a UFD on the public structure is to disable logins and to make sure that no other user is currently logged in. You should reorder a UFD on a private disk only when you are sure that no other job will access the account(s) being reordered on that disk. If you follow the second step, this is no problem.

11.2.4 Error and Processing Messages

During the reordering of a UFD, the REORDR program holds the reordered UFD in a temporary file. At the conclusion of the ordering process, REORDR copies the temporary file back to the original UFD. As REORDR processes each UFD, it generates either a message indicating that the reorder worked or one that tells why an attempted reorder did not work. The messages REORDR prints in the attempt to reorder an account are:

```
Directory dev:[m,n] cannot be Reordered
Directory dev:[m,n] does not exist
Directory dev:[m,n] has been Reordered
Directory dev:[m,n] is not Reordered - File filename.type is OPEN
Directory dev:[m,n] is not Reordered - No write access to UFD
Directory dev:[m,n] is Null
|<-----format---->| |<----------------text-------------------->|
```

The general format and the example of the format contain a device designator and an account number:

```
Directory dev:[m,n] |<-----text------->|
Directory DR3:[1,223] has been Reordered
```

Table 11–3 explains each message.

Table 11–3: REORDR Message and Response

REORDR Message and Response	
cannot be Reordered	Account [0,1] contains open files during timesharing and thus REORDR cannot reorder them.
does not exist	REORDR cannot find the account on the device.
has been Reordered	REORDR has successfully reordered the UFD.
is not Reordered - File filename.typ is OPEN	A user is accessing the UFD and thus REORDER leaves the UFD intact.
is not Reordered - No write access to UFD	REORDER encounters a UFD to which it cannot gain write access. It prints the error message and continues.
is Null	No UFD exists for the account. If REORDR encounters an account without a UFD, it prints the IS NULL error message.

Fatal errors cause REORDR to print a message of the form:

```
?REORDR Fatal Error [code] |<----message----->| AT LINE nnnnnn
```

All fatal REORDR errors include the ?REORDR Fatal Error prefix, a program error code, a message, and a program line number. Code represents the number of the program error that caused the fatal condition to occur. (DIGITAL uses the code number to determine the reason for the fatal error.) REORDR prints a brief message to help identify the error. Finally, the error message includes the line number at which the program came to an end. Submit a Software Performance Report (SPR) to DIGITAL whenever REORDR encounters a fatal error.

11.2.5 REORDR Example

The following example shows the REORDR dialogue and the messages that result from the reordering process:

```
$ RUN AUXLIB$:REORDR
REORDR V9.0 RSTS V9.0 EDERE

Sort Directory(s) (YES/NO) <NO>? YES

Order by CRE[ATION] or ACC[ESS] Date<CRE>? CREATION

In FOR[WARD] or REV[ERSE] Order<FOR>? REVERSE
```

```
Device and UFD Specification(s)? DB1:[*,*],DB2:[*,*]
Directory DB1:[0,1] cannot be Reordered
Directory DB1:[1,1] cannot be Reordered
Directory DB1:[1,3] has been Reordered
Directory DB1:[1,4] has been Reordered
Directory DB2:[1,8] has been Reordered
Directory DB1:[1,13] has been Reordered
Directory DB2:[1,44] is Null
Directory DB2:[1,77] is Null
Directory DB1:[1,200] has been Reordered
Directory DB1:[1,201] has been Reordered
Directory DB1:[1,202] has been Reordered
Directory DB2:[2,227] has been Reordered
Directory DB2:[2,240] has been Reordered
Directory DB2:[26,12] has been Reordered
Directory DB2:[100,100] has been Reordered
Directory DB1:[120,50] has been Reordered
Directory DB2:[120,54] is Null
Directory DB2:[210,203] has been Reordered
Directory DB2:[232,15] has been Reordered
Directory DB2:[232,13] has been Reordered
Directory DB2:[251,0] has been Reordered

Sort Directory(s) (YES/NO) <NO>? <CTRL/Z>
```

11.3 Caching Control

Data caching stores blocks from a user file for direct memory access. Directory caching stores the Master File Directory (MFD), the Group File Directory (GFD), the User File Directory (UFD), monitor overlay code, and other frequently accessed system files and directories.

When a user job executes a read request, the RSTS/E monitor performs a disk access and transfers the requested block(s) of data and directory information from the disk to the user job's buffer. With data and directory caching, the monitor stores the most recently read data blocks in an area of memory called the cache. If a user job executes a read request for either data and/or directory information in the cache, the monitor copies the requested information directly from the cache into the job's buffer. Because the system can thus retrieve data blocks from memory instead of disk, physical disk access is decreased. This can result in improved I/O throughput and faster response time.

Data caching is only useful for read operations. Every write operation causes an actual write to the disk. In a write operation that modifies existing cached data, the data is updated, both in the cache and on the disk.

Both data and directory caching are enabled during system installation. (See the *RSTS/E System Installation and Update Guide.*)

You can disable and reenable data caching with the DCL command SET CACHE. Moreover, DCL commands let you specify caching for a particular file because it lets you mark the file's UFD entry.

Under the BASIC-PLUS or MACRO programming languages, you can use MODE values in the OPEN statement or monitor directives to open a file for cached data access. However, the use of caching MODE values and monitor directives requires TUNE privilege. As system manager, you can use the DCL command SET FILE (see Chapter 12) to designate a file for caching by marking its UFD entry. Once you mark a file's UFD entry for caching, it is cached on OPEN regardless of the user's privilege, as long as caching is enabled on the system. See the *RSTS/E Programming Manual* for information on the BASIC-PLUS OPEN statement and the *RSTS/E System Directives Manual* for information about MACRO directives.

When you enable data caching, the cache receives all data transfer requests that are otherwise directed to the disk driver. Read operations on data that is in the cache occur without placing a load on the disk driver. The monitor constantly updates the cache so that it contains the most recently requested data for cached files.

11.3.1 Size of the Cache

The RSTS/E monitor allocates space for the cache from the Extended Buffer Pool (XBUF). You specify the amount of this allocation with DCL command qualifiers that set the size of a cache cluster (/CLUSTERSIZE=n) and the number of clusters in the cache (/MAXIMUM=n, /DIRECTORY_MAXIMUM=n, and /DATA_ MAXIMUM=n). The size of a cache cluster (1, 2, 4, or 8 blocks) determines the amount of data that is treated as a unit in a read request and, in many cases, the number of read requests that can be resolved in the cache before access to the disk driver is required.

For example, when the cache cluster size is eight blocks, any read operation that installs data in the cache causes eight physically contiguous blocks (including the requested blocks) to be installed. To ensure that only requested data is read in to the cache, make the cache cluster size equal to or less than the pack cluster size (set during disk initialization) of the disk with the most files to be cached.

11.3.2 Sequential and Random Caching Modes

You can cache the data in a file in two ways: sequential or random. You can specify either of these modes for a particular file with the SET FILE command qualifiers (see Chapter 12).

If a job executes a read operation on a cached file and the data is not in the cache, that data is installed in the cache if there is free space or if a current cache cluster is eligible for replacement.

A cluster is eligible for replacement if either of the following conditions is in effect:

- The last block of the cluster has been read in sequential mode.
- The cluster has been in the cache without being read for more than the minimum residency time, which is set using the SET CACHE command.

11.3.3 Random Mode Caching

DIGITAL recommends caching a file in random mode if the file is RMS index structured or a heavily overlaid program. With an RMS indexed file structure, the monitor does not access the data in the file sequentially but must check the file index each time it gets more data. Thus, it reads the index, the data, and the index again.

As an example of random cache operation, consider a read operation executed on a file whose UFD entry is marked for random caching. When a read on the cached file occurs, the monitor examines the contents of the cache to determine if the requested data is present. If the data is in the cache, the data is copied from the cache cluster that contains it. The data is made available to the program and the cache cluster time of last access is updated.

If the requested data is not in the cache, the monitor first attempts to allocate more of XBUF to install the new data (cache limits permitting). If XBUF cannot accommodate more data or the request exceeds the caching limits set with SET CACHE, then the monitor examines the list of cache clusters to find one that is eligible for replacement. If a cluster is not available, the monitor cannot install the requested data in the cache; a normal disk read is automatically performed. If the cache cluster is eligible for replacement, the monitor installs the requested data in that cache cluster and makes the data available to the program.

11.3.4 Sequential Mode Caching

DIGITAL recommends caching a file in sequential mode if you access the contents of the file sequentially. As an example of sequential cache operation, consider a read operation executed on a file whose UFD entry is marked for sequential caching. When a read on the cached file occurs, the monitor examines the contents of the cache to determine if the requested data is present. If the data is in the cache, the data is copied from the cache cluster that contains it. The data is made available to the program. If, in the process, the last block of a cache cluster was read, that cluster is made available for replacement in the cache.

Furthermore, when you request a read operation on any block of a cache cluster (except the last block), the monitor installs a full cluster of data in the cache. That is, if you use the SET CACHE command qualifier /CLUSTERSIZE=n to set a cache cluster size of 8 blocks and then execute a read on the first block of a file, the requested data plus the next 7 blocks from the disk are read into the cache. Thus, the contents of the next 7 reads can be satisfied from the cache. When the last block of the cache cluster is read, the cluster is immediately made available for the installation of new data as requested by other read operations.

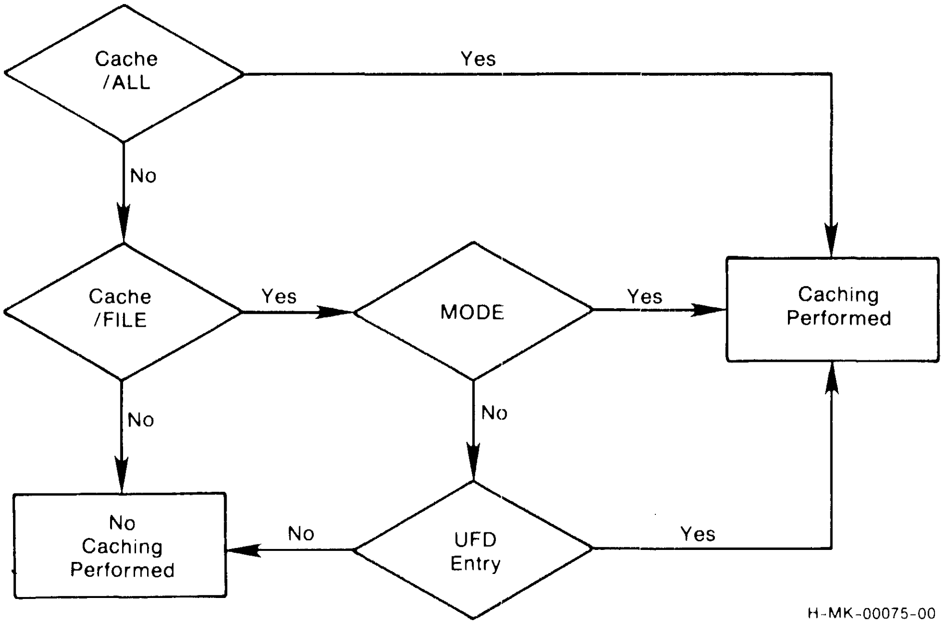
When a read operation from a sequentially cached file causes more than one cache cluster to be read, all of the requested data blocks are made available to the program. However, no data is installed in the cache for any cluster whose last block was read. (That is, only the last cluster can be installed and only if it was incompletely read.) If the cache cluster size is defined as one block and sequential mode is specified, no data blocks are installed in the cache. Thus, every data block is the last block in a cache cluster.

11.4 Caching Guidelines

The relationship between cache cluster size, which you set with a SET CACHE/CLUSTERSIZE=n command, and the pack cluster size of the file can have an effect on the efficiency of caching. The cache cluster size determines the number of blocks on a disk copied into the cache when a cluster is installed. If the cache and pack cluster sizes are equal, there is direct correspondence between the data that will probably be read and the data copied from the disk to the cache. If the cache cluster size is larger than the pack cluster size, the data you request plus some additional (and possibly unrelated) data is copied to the cache, wasting cache clusters. If the cache cluster size is less than the pack cluster size, a read request may require multiple disk accesses to move all of the data into several different cache clusters.

Figure 11–1 shows the flow of UFD entry and MODE value checks performed by the monitor. The result of these checks determines the type of caching to be used on the file.

Figure 11–1: Caching Mode Checks



The size relationship between the cache cluster and RMS indexed file bucket is similar to the size relationship between the cache cluster and the pack cluster. Efficient caching is obtained when the cache cluster and bucket sizes are equal because RMS always reads an entire bucket at once.

Consider the following caching guidelines:

- When setting defaults, the amount of XBUF you allocate to data and directory caching should be as large as possible while maintaining sufficient memory for user jobs, resident libraries, and run-time systems. However, an exceptionally large allocation can increase swapping activity, especially on systems with less than 256K words of memory.
- On systems that use RMS heavily, consider the use of the RMS resident library and the library's effect on XBUF allocation. Typically, on systems with less than 256K words of memory, the resident library should receive priority in memory allocation, to the extent of limiting XBUF allocation. In such a case, data caching may become ineffective and you probably should disable it.

- To optimize directory caching, set all UFD cluster sizes to 16 blocks if possible.

- Files that are frequently accessed sequentially (including RMS sequential and relative files and heavily overlaid programs) should have their UFD entries marked for sequential caching with the DCL command SET FILE/CACHE=SEQUENTIAL.

- Files that are frequently accessed randomly (including RMS relative and indexed files) should have their UFD entries marked for random caching with the DCL command SET FILE/CACHE=RANDOM. Because of the way RSTS/E maintains RMS indexed files, they should always be flagged as random even if they are accessed sequentially. Where possible, RMS indexed file bucket sizes should be a power of two to correspond with caching cluster sizes.

- Use data caching judiciously on RMS indexed files. You gain the most benefit from caching an indexed file when it is subject to many read requests and is open for long periods of time. In addition, you should allocate more XBUF to compensate for the lower caching "hit" rate (the probability that the data you want to read is in the cache) on indexed file data. The hit rate is generally lower on indexed files than it is on sequential files or directory blocks.

- Where possible, all files should be contiguous to reduce window turning.

  Where contiguous files are not possible, you should calculate the file cluster size as follows:

  1. Divide the file size by seven

  2. Round the result to the next higher power of two (not greater than 256 nor less than the pack cluster size)

  NOTE

  If you preextend the file and specify a cluster size of zero, RSTS/E does this calculation for you.

- The minimum residency time is meant to keep some useful data in the cache, even when the cache is very small. Minimum residency time is also meant to make sure that if you want to access previously accessed data the system will find it in the cache, rather than having to read information off disk each time you request more. For systems with large caches or rapidly changing disk access patterns, it may be advantageous to reduce the minimum residency time from its default value of 60 seconds. Use the /KEEP=n qualifier to modify minimum residency time.

Caching is not the most efficient data access method for all files under all circumstances. But, in general, a high-access file is a logical candidate for caching. A high-access file meets one or more of the following conditions:

- Accessed by more than one user at a time
- Opened for long periods of time
- Frequently accessed
- Accessed sequentially (indexed files)
- Primarily read
- Is an overlaid program (such as a user program, TKB, or DTR)

However, because of the large number of different applications, you must experiment to arrive at the best use of caching on your system.

11.5 Virtual Disks

You can create a virtual disk that acts like a physical disk, only faster. Use the DCL commands CREATE/VIRTUAL_DISK and DELETE/VIRTUAL_DISK to create and delete the virtual disk. You must have the HWCFG and INSTAL privileges to use these commands. A system can have no more than one virtual disk.

The virtual disk speeds up I/O, but it uses up a great deal of CPU time. Do not create a virtual disk on a system that is short of CPU time. Use the virtual disk when I/O is your principal bottleneck.

You can create a virtual disk at any time, but it takes up a large amount of memory. Therefore, DIGITAL strongly recommends that you create the virtual disk immediately after booting, before any libraries or auxiliary run-time systems get loaded. This minimizes the amount of time spent swapping things in and out of memory at the time of creation.

11.6 DCL Command Descriptions

The following sections describe the DCL commands and qualifiers for Device Management and Caching Control.

11.6.1 ALLOCATE Command

The ALLOCATE command allocates a device and optionally assigns it a logical name. ALLOCATE does not require any privileges to allocate a device to your own job.

Format

ALLOCATE *dev: [logical_name[:]]*

Command Qualifiers	Defaults
/JOB=n	none
/SEIZE	none

Prompts

Device:

Command Parameters

dev:
Specifies the device to be allocated.

logical_name[:]
Specifies the logical name to be assigned to the device.

Command Qualifiers

/JOB=n
Allocates the device to the specified job. You need HWCTL privilege to use this qualifier.

/SEIZE
Reallocates a device currently allocated to another job. The /SEIZE switch returns an error message if a file is open on the device. You need HWCTL privilege to use this qualifier.

11.6.2 CREATE/VIRTUAL_DISK Command

The CREATE/VIRTUAL_DISK command reserves a section of memory for use as the virtual disk. Once the virtual disk has been created, you can initialize and mount it using the INITIALIZE and MOUNT commands. Use the SHOW MEMORY command to verify the location of the virtual disk after it has been created. Note that the virtual disk appears as locked out in the memory list until it has been initialized and mounted. You need the INSTAL and HWCFG privileges to use the CREATE/VIRTUAL_DISK command. Example:

```
$ CREATE/VIRTUAL_DISK 38
Virtual Disk created with size of 40 blocks
$ INITIALIZE/EXERCISE=0 DV0: V
$ MOUNT DV0: V
$ SHOW MEMORY

Memory allocation table:
Start   End  Length  Permanent   Temporary
  0K -   99K ( 100K) MONITOR
100K -  102K (   3K) MONITOR
103K -  426K ( 324K)   (User)
427K -  436K (  10K) Virtual Disk
437K -  446K (  10K)   (User)    DAPRES LIB
447K -  450K (   4K)   (User)    RMSRES LIB
451K -  511K (  61K) ** XBUF **
512K -  *** END ***
$ SHOW DISK

Disk Structure:
Dsk  Open    Size     Free    Clu    Err Name       Level  Comments
DV0     0      40       28  70%  1      0 V           1.2   Pri, DLW
DU1    18   83204    47718  57%  2      0 RD32        1.2   Pub, DLW
```

Format

CREATE/VIRTUAL_DISK *disk_size*

Command Qualifiers	Defaults
/ADDRESS[=address]	highest available memory
/[NO]LOG	/LOG

Prompts

Virtual disk size:

Command Parameters

disk_size

Specifies the size, in blocks, of the virtual disk. If the disk size is not a multiple of 4, the system rounds it up to the next multiple of 4 blocks. A confirmation message tells you the actual size of the virtual disk.

Command Qualifiers

/ADDRESS[=address]

Specifies the address, in K-words, at which to locate the virtual disk. The address must be between 1 and 2043. Use the SHOW MEMORY command to select an available address to locate the virtual disk. The system returns an error message if the address is above the memory on the system, or if any part of the memory is in use.

If you do not specify an address, the system puts the the virtual disk at the highest section of memory in which it will fit. The system returns an error message if there is not enough available memory to place the disk.

/[NO]LOG

Specifies whether or not you get a confirmation message saying the virtual disk was created. The confirmation message shows the actual size of the virtual disk. The default is /LOG.

11.6.3 DEALLOCATE Command

The DEALLOCATE command deallocates the specified device. DEALLOCATE does not require any privileges.

Format

DEALLOCATE *dev[:]*

Command Qualifiers	Defaults
/ALL	none

Prompts

Device:

Command Parameters

dev:
Specifies the device to be deallocated.

Command Qualifiers

/ALL
Deallocates all devices that were allocated to the current job.

11.6.4 DELETE/VIRTUAL_DISK Command

Frees up the section of memory reserved for the virtual disk. To delete the virtual disk, you must dismount it. It must not have files opened on it or an error will be returned. You need the INSTAL and HWCFG privileges to use the DELETE/VIRTUAL_DISK command. Example:

```
$ DELETE/VIRTUAL_DISK
DELETE/VIRTUAL_DISK
^
?Virtual disk is mounted or has open files
$ SHOW DISK

Disk Structure:
Dsk  Open    Size      Free     Clu    Err Name        Level  Comments
DV0     0      40        28  70%   1      0 V             1.2  Pri, DLW
DU1    18   83204   47718  57%   2      0 RD32          1.2  Pub, DLW

$ DISMOUNT DV0: V
$ DELETE/VIRTUAL_DISK
Virtual disk deleted
```

Format

DELETE/VIRTUAL_DISK

Command Qualifiers	Defaults
/[NO]LOG	/LOG

Prompts

none

Command Qualifiers

/[NO]LOG

Specifies whether or not the system displays a confirmation message telling you the requested action was taken. The default is /LOG.

11.6.5 SET CACHE Command

The SET CACHE command lets you set caching parameters for the system as a whole. The SET CACHE command lets you enable data and directory caching on the system. Note that commands in the system startup command file normally enable caching automatically. You need TUNE privilege to use the SET CACHE command.

When you use SET CACHE with no qualifiers, the last settings specified are applied. The initial system defaults are:

/CLUSTERSIZE	4 blocks
/MAXIMUM	No limit on total cache allocation
/DIRECTORY_MAXIMUM	No limit on directory caching
/DATA_MAXIMUM	No limit on data caching
/KEEP	60 seconds

If you use SET CACHE/ALL, all read requests are cached. The type of caching (sequential or random) for a particular file is determined by the file's UFD entry as specified in the SET FILE command or by an OPEN MODE specification. DIGITAL recommends that you avoid using SET CACHE/ALL. It generally reduces performance except for a small number of specific applications.

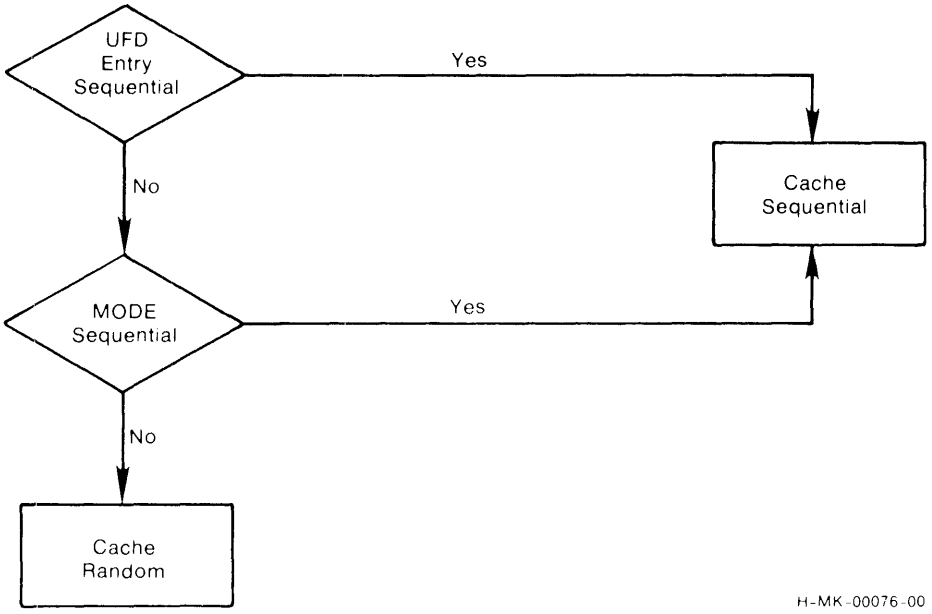
If you use SET CACHE/FILE, a file is cached on the basis of its UFD entry and/or the specified OPEN MODE. That is, a file is cached if either the UFD entry or OPEN MODE specifies caching; it is cached sequentially if either the UFD entry or OPEN MODE specifies sequential.

If you use SET CACHE/NOFILE, RSTS/E does not cache files. This qualifier is used to enable directory caching but disable data caching on the system.

Data and directories on a virtual disk (DV:) are never cached, since they are already in memory.

Figure 11–2 shows the mechanism by which the monitor decides the amount of caching to be performed. The figure shows the flow of DCL commands and caching specifications examined by the monitor.

Figure 11–2: Monitor Caching Checks



Use the SET CACHE/CLUSTERSIZE=n command to specify the cache cluster size as 1, 2, 4, or 8 blocks for each cluster. The cache cluster size controls the number of contiguous data blocks that are copied from disk to the cache whenevei file data is cached. The cache cluster size should be small enough to contain only the target blocks but large enough to reduce the number of disk accesses. This means you must anticipate data requests and ensure that the cache cluster size is equal to the file cluster size of the files most frequently accessed.

The SET CACHE/MAXIMUM=n command (where n is in the range 0 to UNLIMITED) specifies the total number of cache clusters used by both directory and data caching. The default limit is UNLIMITED, which means that cache cluster allocation is limited only by the amount of XBUF available.

The SET CACHE/DIRECTORY_MAXIMUM=n and SET CACHE/DATA_MAXIMUM=n commands (where n is in the range 0 to UNLIMITED) specify the cache cluster allocation for directory and data caching, respectively. The specified allocation is an upper limit. Thus, if a 40K-word XBUF is defined at system installation and /DIRECTORY_MAXIMUM and /DATA_MAXIMUM both specify a number of cache clusters equivalent to 25K, data can use space in the cache up to a maximum of 25K, which leaves a 15K minimum for directory caching. The reverse is also true. In this manner, data and directory caching are guaranteed a minimum allocation and the amount of overlap is controlled, which permits the cache to dynamically adjust to system and program requirements.

The SET CACHE/KEEP=n command specifies the cache replacement time in seconds. This is the amount of time that RSTS/E keeps an unused cache cluster in memory before the monitor makes it elligible for replacement by another cache cluster.

The SET CACHE/DISABLE command disables all data and directory caching on the system. If you reenable caching following a disable and do not specify caching parameters (/CLUSTERSIZE, /MAXIMUM, /DIRECTORY_MAXIMUM, /DATA_MAXIMUM, and /KEEP), the default parameters are those that were in effect before the disable.

Format

SET CACHE

Command Qualifiers	Defaults
/ALL	none
/CLUSTERSIZE=n	/CLUSTERSIZE=4
/DATA_MAXIMUM=n	none
/DIRECTORY_MAXIMUM=n	none
/DISABLE	/ENABLE
/ENABLE	/ENABLE
/[NO]FILE	/FILE
/KEEP=n	none
/MAXIMUM=n	none

Prompts

none

Command Qualifiers

/ALL
Caches all data transfers.

/CLUSTERSIZE=n
Specifies a cache cluster size of n with values equal to 1, 2, 4, or 8 blocks. The default is /CLUSTERSIZE=4.

/DATA_MAXIMUM=n
Specifies the maximum number of clusters (based on available memory) used for data caching in the range 0 to 65535 or the keyword UNLIMITED.

/DIRECTORY_MAXIMUM=n
Specifies the maximum number of cache clusters used for directory caching in the range 0 to 65535 or the keyword UNLIMITED.

/DISABLE
Disables data and directory caching.

/ENABLE

Enables caching on the system using the previous settings as the defaults. The default is /ENABLE.

/[NO]FILE

Indicates if file data should be cached. The /FILE qualifier indicates that file data is cached depending on their file attributes or OPEN MODE. The /NOFILE qualifier indicates that no file data is cached. The default is /FILE.

/KEEP=n

Specifies the cache replacement time in seconds that an unused cache cluster is kept in memory before the space can be used for another cache cluster. The value of n is in the range 0 to 65535.

/MAXIMUM=n

Specifies the maximum number of cache clusters used for directory and data caching in the range 0 to 65535 or the keyword UNLIMITED.

11.6.6 SET DEVICE Command

The SET DEVICE command sets the characteristics for the specified device. You need HWCTL privilege to use /ENABLE, HWCFG to use /RESTRICT, HWCFG and HWCTL to use /DISABLE, and HWCFG and DEVICE to use /NORESTRICT.

Format

SET DEVICE *dev:*

Command Qualifiers	Defaults
/DISABLE	none
/ENABLE	none
/[NO]RESTRICT	none

Prompts

Device:

Command Parameters

dev:
Specifies the device whose characteristics you want to change.

Command Qualifiers

/DISABLE
Disables the specified device. No other qualifiers can be specified with /DISABLE.

/ENABLE
Enables a device that has been disabled by the SET DEVICE/DISABLE command.

/[NO]RESTRICT
Tells the system which users can open or allocate the specified device. The /RESTRICT qualifier lets only users with DEVICE privilege open or allocate the device. The /NORESTRICT qualifier lets any user open or allocate the device.

11.6.7 SET PRINTER Command

The SET PRINTER command sets the characteristics of the specified line printei You need HWCFG privilege to use the SET PRINTER command.

Format

SET PRINTER *dev[:]*

Command Qualifiers	Defaults
/BACKSPACE[=mode]	/BACKSPACE=REAL
/[NO]CONTROLS	none
/[NO]CR_PREFIX	none
/DEVICE_TYPE=type	none
/[NO]EIGHT_BIT	none
/[NO]EOT	none
/[NO]FILL	none
/LOWERCASE	none
/[NO]OMIT_CR	none
/PAGE_LENGTH=n	none
/[NO]SPECIAL_CHARACTER[=char]	none
/[NO]TAB	none
/UPPERCASE	none
/WIDTH	none

Prompts

Printer:

Command Parameters

dev[:]

Specifies the line printer whose characteristics you want to set.

Command Qualifiers

/BACKSPACE[=mode]

Tells the system how to handle backspace characters for this printer. The possible modes are:

- CONTROL—Treats a backspace like other nonprinting characters.
- REAL—Uses the actual backspace feature on the specified printer.
- SIMULATE—Simulates the action of backspace on the specified printer by issuing a carriage return (CR) followed by the appropriate number of spaces to position the print head one column to the left of its original position.

The default is /BACKSPACE=REAL.

/[NO]CONTROLS

Tells the system how to handle nonprinting characters. /CONTROLS sends nonprinting characters to the line printer. /NOCONTROLS discards nonprinting characters or uses up arrow mode.

/[NO]CR_PREFIX
Tells the system how to handle line feed (LF), vertical tab (VT), and form feed (FF) characters. The options are:

- /CR_PREFIX inserts a CR before a LF, a VT, or a FF.
- /NOCR_PREFIX tells the printer to do an implied CR before a LF, a VT, or a FF.

/DEVICE_TYPE=type
Sets the characteristics of the line printer to the specified type:

- LA180—Sets the characteristics of the line printer to those of an LA180.
- LN01—Sets the characteristics of the line printer to those of an LN01.
- LP11—Sets the characteristics of the line printer to those of an LP11.

When you use /DEVICE_TYPE, you automatically set all the other printer characteristics as well.

/[NO]EIGHT_BIT
Tells the system how to handle 8-bit characters. The /EIGHT_BIT qualifier lets you send 8-bit characters to the line printer. The /NOEIGHT_BIT qualifier does not let you send 8-bit characters to the line printer.

/[NO]EOT
Tells the system how to handle an end-of-transmission character. The /EOT qualifier sends an end-of-transmission (Ctrl/D) to the line printer. The /NOEOT qualifier treats an end-of-transmission (Ctrl/D) like other nonprinting characters.

/[NO]FILL
Tells the system how to handle fill characters. The /FILL qualifier inserts fill characters after form feeds. The /NOFILL qualifier does not insert fill characters after form feeds.

/LOWERCASE
Sends lowercase characters to the line printer.

/[NO]OMIT_CR
Instructs the system on how you want the CR character handled. The /OMIT_CR qualifier tells the system to omit the CR if the next character is a LF. The /NOOMIT_CR qualifier tells the system to always send CR to the printer.

/PAGE_LENGTH=n
Sets the form length of the line printer to n. The form length can be in the range 1 to 255.

/[NO]SPECIAL_CHARACTER[=char]
Instructs the system on how you want the printer to handle its special character. The /SPECIAL_CHARACTER=char qualifier tells the printer to print its special character whenever it receives the specified char. Char can be given by its numeric value in the range 0 to 255 or by enclosing the character in quotation marks (") if it is printable. The /NOSPECIAL_CHARACTER qualifier tells the system to disable the special character feature for the specified line printer.

/[NO]TAB
Instructs the system on how you want the TAB character handled. The /TAB qualifier indicates that the printer has hardware TAB capability. The /NOTAB qualifier indicates that the printer does not have hardware TAB capability and the system has to translate TAB characters to the appropriate number of spaces.

/UPPERCASE

Tells the system to convert lowercase characters to uppercase before sending them to the line printer.

/WIDTH=n

Sets the width of the line printer to n. The width can be in the range 0 to 254.

11.6.8 SHOW CACHE Command

The SHOW CACHE command displays the current caching parameters for your system. SHOW CACHE does not require any privileges.

The listing includes information on cluster sizes, and data caching. For example:

```
$ SHOW CACHE
Caching configuration
---------------------
   Caching enabled for Directories and Data
   Cache Keep timer = 60 seconds
   Cluster size = 4
   Data caching = File selectable
XBUF usage (in K-words)
-----------------------
   Total cache:            229.00 (maximum clusters allowed: Unlimited)
   Directory cache:        200.00 (maximum clusters allowed: Unlimited)
   Data cache:              29.00 (maximum clusters allowed: Unlimited)
   ASSIGN/SYSTEM tables:     0.94
   LOAD/OVERLAY tables:      3.16
   LOAD/INDEX tables:        9.22
   DCL Buffers:              1.13
   Quota tables:             0.38 (small buffers in use: 0)
   Window turn blocks:       0.00 (maximum blocks used: 5)

$
```

In this example, SHOW CACHE reports that:

- Directory and data caching are currently enabled
- The cache keep timer is set to 60 seconds
- The cache cluster size is 4 blocks
- Only selected files are cached
- No limit is set on the use of XBUF for directory and data caching
- 229 K words of XBUF are currently being used for directory and data caching
- 200 K words of XBUF are currently being used for directory caching
- 29 K words of XBUF are currently being used for data caching
- 0.94 K words of XBUF are currently being used to store system logicals (ASSIGN/SYSTEM)
- 3.16 K words of XBUF are currently being used to store monitor overlays (LOAD/OVERLAY)
- 9.22 K words of XBUF are currently being used to store disk allocation indexes (LOAD/INDEX)
- 1.13 K words of XBUF are currently being used for DCL buffers
- 0.38 K words of XBUF are currently being used to store logged-in disk quota tables
- No small buffers are being used to store logged-in disk quotas
- No XBUF is currently being used for storing file window turn blocks
- A maximum of 5 window turn blocks have been used since the system was started

Format

SHOW CACHE

Command Qualifiers	Defaults
none	none

Prompts

none

11.6.9 SHOW DEVICE Command

The SHOW DEVICE command displays information about the specified device. If you do not specify a device, RSTS/E displays information about all devices. SHOW DEVICE does not require any privileges.

Format

SHOW DEVICE *[dev[:]]*

Command Qualifiers	Defaults
/ALL	/ALL
/ALLOCATED	

Prompts

none

Command Parameters

dev[:]
Specifies the device whose characteristics you want to display.

Command Qualifiers

/ALL
Displays information about all devices configured on the system. A device specification overrides the /ALL qualifier.

/ALLOCATED
Displays information about all devices that are currently open or allocated. The display includes devices that are open as non-file-structured devices.

11.6.10 SHOW DISKS Command

The SHOW DISKS command displays the status of all mounted disks. SHOW DISKS does not require any privileges.

Format

SHOW DISKS

Command Qualifiers	Defaults
none	none

Prompts

none

11.6.11 SHOW PRINTER Command

The SHOW PRINTER command displays the characteristics of the specified printer. SHOW PRINTER does not require any privileges.

Format

SHOW PRINTER *dev[:]*

Command Qualifiers	Defaults
/ALL	/ALL

Prompts

none

Command Parameters

dev:

Specifies the printer whose characteristics you want to display.

Command Qualifiers

/ALL

Displays information about all the printers configured on the system. A printer specification overrides the /ALL qualifier.

Chapter 12

Managing System Files and Logical Names

This chapter describes the DCL commands that you use to manage RSTS/E system files and logical names. Table 12–1 lists and briefly describes the DCL commands available to you. The rest of the chapter lists each command alphabetically and explains it in more detail.

Table 12–1: DCL Commands for Managing System Files and Logical Names

Command	Meaning
Run Time System Control	
INSTALL/RUNTIME_SYSTEM	Adds the named run-time system to the list of installed run-time systems.
REMOVE/RUNTIME_SYSTEM	Removes the name of a run-time system module as an entry in the run-time system table.
SHOW RUNTIME_SYSTEM	Displays information about all installed run-time system.
UNLOAD/RUNTIME_SYSTEM	Unloads the named run-time system from memory.
Resident Library Control	
INSTALL/LIBRARY	Adds the named library to the list of installed resident libraries.
REMOVE/LIBRARY	Removes the named file as a resident library.
SHOW LIBRARY	Displays information about all installed libraries.
UNLOAD/LIBRARY	Removes the named resident library from memory.
System Logical Names Control	
ASSIGN/SYSTEM	Assigns a system-wide logical name to the specified device type and unit. If you include an account number, the ASSIGN/SYSTEM command also associates that PPN with the logical name.

(continued on next page)

Table 12–1 (Cont.): DCL Commands for Managing System Files and Logical Names

System Logical Names Control	
DEASSIGN/SYSTEM	Removes the entire entry for the logical name from the system logical table. It can also remove logical names for devices on your system. Device mnemonic names for system devices have permanent entries in the logical table. If you use the DEASSIGN/SYSTEM command to remove a logical name associated with a device, only the logical name goes away, not the device mnemonic. You can then assign a new name to that device.
SHOW LOGICAL/SYSTEM	Displays information about a system-wide logical name.
System wide Command Language Control	
DEFINE/COMMAND/SYSTEM	Creates or redefines a system-wide command.
DELETE/COMMAND/SYSTEM	Deletes the specified command from the list of defined system-wide commands.
SHOW COMMAND/SYSTEM	Displays information about the specified system-wide command.
System File Control	
DUMP/SYSTEM	Records in the CRASH.SYS file the current state of the system so you or field service has a way to investigate reasons for system malfunctions.
INSTALL/ERROR_FILE	Installs the specified file as the system error message file.
INSTALL/OVERLAY_FILE	Installs the specified file as the system overlay file.
INSTALL/SWAP_FILE=n	Adds the specified file to the list of installed swap files. Valid values of swap-index are 0, 1, or 3.
REMOVE/ERROR_FILE	Removes the system error message file.
REMOVE/OVERLAY_FILE	Removes the system overlay file.
REMOVE/SWAP_FILE=n	Removes the specified swap file from the list of installed swap files. Valid values of swap-index are 0, 1, or 3.
SHOW FILE	Displays information about the specified file(s).
User File Control	
SET FILE	Sets the file characteristics.
SHOW LOGICALS	Displays information about a user-defined logical name.

12.1 Run-Time System Control

A run-time system on RSTS/E is the common, shareable part of a user job and, in some cases (for example, BASIC-PLUS), is the interface between the user's executable code and the monitor. Run-time systems control execution of jobs when the monitor allows the jobs to run. The monitor sets up jobs and establishes an environment in which a run-time system can function. The BASIC-PLUS run-time system interprets requests made by a user job, translates those requests into a format the monitor understands, and transmits the requests of the monitor. The monitor processes the requests and passes either data or an error to the job. The run-time system decides whether to interpret the data or log an error. Jobs running under the BASIC-PLUS run-time system cannot make direct requests to the monitor.

Other run-time systems may not be language interfaces in the same sense that the BASIC-PLUS run-time system is. Jobs under their control can make direct requests of the monitor. In such cases, the major role of the run-time system is to load compiled programs to handle system errors, and return control, on program termination, to the job's keyboard monitor. Some of these run-time systems can also emulate the action of certain system directives from other operating systems, such as RT11.

A run-time system interprets user requests when it is a language interface for a user job. A computer language provides a standard means by which many users can define data, process it, and obtain results. The language relieves each user of the need to learn the workings of the monitor. The run-time system under which the language operates is more efficient because multiple users can share the same code. This ability to share code is more efficient for the system because each user need not have a copy of the common code.

All RSTS/E installations have DCL as the default keyboard monitor and can optionally have auxiliary run-time systems. DCL:

- Allows users access to system resources
- Interprets system command requests
- Performs housekeeping chores

An auxiliary run-time system:

- Provides additional shared computer processing functions
- Enables processing in other languages

Such an auxiliary run-time system may compile and/or execute BASIC-PLUS-2, COBOL, DIBOL, FORTRAN-IV, FORTRAN-77, or other language operations.

The system account [0,1] stores run-time systems as contiguous files with .RTS file types. You must store DCL on the system disk because, at the start of timesharing, the system disk is the only disk mounted on the system. You can store auxiliary run-time systems on any disk, either public or private. (Auxiliary run-time system files can also be stored in accounts other than [0,1].) All run-time system files are contiguous because they must be loaded into memory in the fastest time possible.

Auxiliary run-time systems are created either during system installation or afterwards. The monitor structures that describe these run-time systems do not exist automatically at the start of timesharing. Therefore, you must provide the commands to add monitor structures for the auxiliary run-time systems that you want to make available for any given time-sharing session. The structure for an auxiliary run-time system is dynamic and can be removed during a time-sharing session.

The DCL run-time system commands let you control operations of auxiliary run-time systems. The DCL commands perform additional processing for certain commands. To assist the system programmer, the command descriptions distinguish between the monitor call action and the DCL command action. Because run-time systems under development need certain special controls, some features of run-time system control are intended for Digital personnel only. Nevertheless, all features are described to give you a broader perspective.

Digital recommends that you make auxiliary run-time systems available by commands in the START.COM file. By reading the descriptions in this section, you can understand what actions should occur for each run-time system command.

12.1.1 Adding and Removing Auxiliary Run-Time Systems

Because the structures that define auxiliary run-time systems are temporary, you must create them for each time-sharing session. The INSTALL/RUNTIME_SYSTEM command creates the necessary structure to enable an auxiliary run-time system to function. INSTALL/RUNTIME_SYSTEM command qualifiers can alter predefined characteristics of the run-time system.

The INSTALL/RUNTIME_SYSTEM command creates a run-time system description block with a general small buffer. A file with the name of the run-time system and a file type of .RTS can be in any account on the disk specified in the command. A run-time system with the same name cannot already exist on the system. The system makes sure that the file found is contiguous and has the proper format and parameters.

You can specify qualifiers to override predefined characteristics. The INSTALL/RUNTIME_SYSTEM command extracts information from the run-time system file to establish the proper entries in the run-time system description block. To establish the entries, the monitor opens the file, reads it, and sets up the predefined characteristics to be placed in the description block. You can change the predefined characteristics by the qualifiers: /PREFIX=n, /[NO]UNLOAD, /[NO]ERROR_LOGGING, /[NO]SHAREABLE, /[NO]READ_ONLY, and /[NO]KEYBOARD_MONITOR.

You can also change the predefined characteristics by negating these qualifiers. For example, if you were testing a run-time system with no special command decoding capabilities, you would use the qualifiers /NOSHAREABLE, /NOREAD_ONLY, /NOERROR_LOGGING, and /UNLOAD to add the run-time system. The /NOSHAREABLE makes sure that no other job tries to share the untested code; /NOREAD_ONLY is necessary for setting breakpoints with ODT; /NOERROR_LOGGING prevents false errors generated by the run-time system from cluttering the system error log; /UNLOAD makes sure that the run-time system's image is not left in memory but is reloaded from disk immediately preceding every entry to it.

The qualifiers and negations of the qualifiers do not alter the predefined characteristics in the run-time system file; only the characteristics defined in the description block are altered. The characteristics remain in effect until you remove the run-time system.

The /[NO]ADDRESS[=n] qualifier lets you take advantage of high speed memory on the system or avoid fragmentation of memory. Without the /ADDRESS=n qualifier, the monitor decides where to load the run-time system each time residency is required. With the qualifier, the monitor loads the run-time system at the specific 1K-word section of memory. You must include the /ADDRESS=n qualifier whenever you use the /NOREAD_ONLY qualifier.

The /ADDRESS qualifier without an argument indicates that the monitor should compute a starting address by finding the highest memory segment large enough to hold the run-time system.

The /NOADDRESS qualifier lets the run-time system "float" in memory; that is, RSTS/E computes a new address every time the run-time system is loaded into memory. The default is /NOADDRESS.

Because 1K-word section numbering begins at 0 and ends at n-1 (where n is the total size of memory), the 1K section number in the /ADDRESS=n qualifier is one less than the physical section number. For example, to load the RT11 run-time system (4K words in size) into the 61st through 64th 1K sections of memory, specify /ADDRESS=60. The run-time system is loaded from low memory to high memory at its defined initialized size. To be loaded without error, enough contiguous user space must be available starting at that location.

The location specified in the /ADDRESS=n qualifier becomes the default location at which the run-time system is loaded during the current time-sharing session. You need to change the location only if you exchange the allocation of the section of memory with either the DEFAULT or the START initialization options.

One precaution is necessary when specifying the address at which the run-time system is loaded. The section of memory you choose must not fragment the user job space to prevent the run-time system from executing a job. For example, assume a system has 24K words of user space available between the 36K and 60K sections of memory. Assume also that a job requires 18K words of user space to run and that the run-time system requires 4K words when resident. If the loading address is 36K, the space between 40K and 60K remains available for an 18K job to run. If the loading address is 42K, the user space is fragmented into two sections — one from 36K to 42K and one from 46K to 60K. An 18K-word job area is not available to execute a job using this auxiliary run-time system.

The system verifies that the memory section given in the /ADDRESS=n qualifier is reasonable. If the entire range of memory starting at the load address is not available, the system prints:

```
?Memory not available
```

You should consult a memory status report (for example, the report you get with the SHOW MEMORY command) to select an available range of memory. If the range of memory results in fragmenting that may cause a swapping violation, the system prints the error message:

```
?Memory would be fragmented
```

A swapping violation occurs if the memory to be occupied by the run-time system does not allow enough contiguous space for a maximum sized job to run.

Appendix D contains a summary of the errors that can occur when you are installing a run-time system.

The /POSITION=n qualifier indicates the position in the linked list of blocks in which the run-time system block is to be placed. The default keyboard monitor block is always first in the list. If the run-time system you are adding is to be accessed frequently, you can reduce system overhead by placing it in the next position in the list. Without the qualifier, the system adds the block to the end of the list.

If n in the /POSITION=n qualifier is one, the block is placed immediately after that of the default keyboard monitor. If n is zero or a value greater than the number of currently defined run-time systems, the block is added to the end of the list. The default is /POSITION=0.

The position of a run-time system description block in the linked list affects how the system treats a RUN request for a file without a file type. On receiving such a RUN request, the system checks the indicated directory for all files with the specified name and an executable file type.

For example, if the directory contained three files of the same name with file types .BAC, .TSK, and .SAV, the system checks for the run-time system nearest the default keyboard monitor in the linked list of description blocks. If RT11 was nearer than BASIC, the system runs the .SAV version of the file.

The /TYPE=filetype qualifier on the INSTALL/RUNTIME_SYSTEM command changes the default file type used for an executable file. If you do not specify this qualifier, the monitor extracts the file type defined in the run-time system file. The system applies this default file type when a user types a RUN command for a program and does not specify a file type.

The /MINIMUM=s and /MAXIMUM=s qualifiers alter the minimum and maximum job sizes (in K words) defined in the file for the run-time system. Generally, the minimum value allowed is one and the maximum is 32, the size of RTS, but a given run-time system may not allow this full range of job sizes. For BASIC-PLUS, the minimum and maximum sizes are 2 and 16, respectively.

The /LOCK qualifier makes the run-time system permanently resident. Usually, an auxiliary run-time system is temporarily resident. It occupies memory as long as the currently active job is running under its control. The system frees the memory it occupies when that memory is needed to load another run-time system or to load a job running under another run-time system. (Thus, the run-time system may be nonresident if there are active jobs using it.) The system automatically loads the run-time system when a user needs it. If you specify the /LOCK qualifier, the auxiliary run-time system becomes permanently resident, as the default keyboard monitor is. The only way to make such a run-time system nonresident is to use the DCL command UNLOAD/RUNTIME_SYSYTEM.

The REMOVE/RUNTIME_SYSTEM command reverses the steps the INSTALL/RUNTIME_SYSTEM command performs. Before you can shut down the system, you must remove all auxiliary run-time systems. The SHUTUP program performs this removal operation automatically. The system checks to make sure that no jobs are currently running under control of the run-time system to be removed. If the usage count is zero, the description block is removed from the list and the run-time system file is closed.

The UNLOAD/RUNTIME_SYSTEM command frees the memory occupied by the run-time system. If a job is currently resident or being loaded and intends to use the run-time system, the system prints:

```
?Run-time system in use
```

12.1.2 Associating a File with a Run-Time System

The SET FILE/RUNTIME_SYSTEM=name command changes the name of the run-time system associated with a file. Every disk file on RSTS/E has in its directory the name of the run-time system under which it was created, except for files that are larger than 65,535 blocks. On a RUN request for the file, RSTS/E checks the name to find out what run-time system to use for the job. RSTS/E automatically passes control to that run-time system which, in turn, executes the file.

To change the name, RSTS/E attempts to open the file to establish write access. If write access is denied, it prints:

```
?Protection violation
```

The name of the run-time system is written in the directory of the file specified. RSTS/E performs this naming operation during the system library build procedures when a file designed to run under an auxiliary run-time system is transferred from the distribution medium.

12.2 Resident Library Control

A resident library is a collection of shareable routines or data that the task builder links together into a task image file on disk. The MAKSIL program formats this disk file into Save Image Library (SIL) format. You can store resident libraries on any disk, either public or private. (Resident library files can be stored in accounts other than [0,1].) All resident library files are contiguous because they must be loaded into memory in the fastest time possible. See the *RSTS/E Programmer's Utilities Manual* for more information on resident libraries.

You then use the INSTALL/LIBRARY command to assign the task image portion of the SIL file to a contiguous region of physical memory. Note that you can also use monitor SYS call -18 to assign the task image portion of the SIL file to memory. Once the body of shareable routines or data is linked, formatted, and assigned to memory, it becomes a resident library that is accessible to user tasks as part of their virtual address space.

12.2.1 Resident Library DCL Commands

You can use the DCL commands to install, remove, load, or unload a resident library. For example, the DCL command to add a resident library is:

```
$ INSTALL/LIBRARY library-file-spec
```

You can specify any valid RSTS/E file specification. You can also specify a protection code in addition to the library file-spec, using the /PROTECTION qualifier. If you do not specify a device, then the system defaults to the system disk. The defalt PPN is account [0,1], and the default protection code is <42> (read-only). The file type must be .LIB; otherwise an error occurs.

The /[NO]ADDRESS[=n] qualifier locates the resident library at a specific point in memory. The n variable can be any number from the lowest to the highest available memory locations in increments of 1K-word sections. The /ADDRESS qualifier without an argument indicates that the monitor should compute a starting address by finding the highest memory segment large enough to hold the library.

The /NOADDRESS qualifier lets the library "float" in memory; that is, RSTS/E computes a new address every time it loads the library into memory. However, this method of loading has the following restrictions:

- A program can only map one such library at any time
- The program must be running under the RSX emulation run-time system
- The maximum size of the library is 28K words
- The highest address for mapping the library is 32K

The /LOCK qualifier lets you make a resident library permanently resident. Usually, a resident library remains in memory only temporarily; that is, it occupies memory only as long as currently active jobs run under its control. However, when you need to access the library again, the system automatically reloads it. To force the library to remain permanently in memory, you must specify the /LOCK qualifier in the INSTALL/LIBRARY command.

If you want to remove a resident library from memory, delete the monitor structure that defines the library, and close the library file, use the DCL command:

```
$ REMOVE/LIBRARY name
```

Note that the REMOVE/LIBRARY command makes the resident library inaccessible for sharing. However, the /UNLOAD qualifier, which can be used only with the INSTALL/LIBRARY command, only temporarily removes the library from memory while it is not in use.

The UNLOAD/LIBRARY command removes a resident library from memory and thus frees memory that can then be used to load another library or be available for other system purposes. If you attempt to unload a resident library that is in the process of being loaded or is in use by the currently running job, the system prints the message:

```
?Library in use
```

To prevent inadvertent loss of data, the monitor does not automatically unload a library you have loaded with the /NOREAD_ONLY qualifier. Unloading a library removes it from memory but does not write it back to disk. This means the system discards the library and retrieves a new copy from the disk when it is needed again. Thus, anything you write to a library that you mark read/write is lost after you unload it with the UNLOAD/LIBRARY command. You must use the UNLOAD/LIBRARY command to unload a resident library that is designated read/write.

Whether a library remains in memory depends on the qualifiers you attach to the INSTALL/LIBRARY command. The /LOCK and /NOREAD_ONLY qualifiers prevent the monitor from removing it automatically. In addition, the monitor does not remove a library to which a user job is attached; instead, the system prints:

```
?Library in use
```

12.3 System Logical Names

RSTS/E lets users access devices by logical names as well as by physical names. Logical names that apply to all users are called system logical names. On all systems, users can refer to a disk by its pack identification or by a name that replaces the pack identification. Thus, each disk unit configured on RSTS/E systems has the capability of being accessed by a system logical name. Logical names that apply to a single job are called user logical names.

You can define system logical names for nondisk devices and additional names for disk devices. The number of names is limited by the size of XBUF. Such a system logical name can have an account number associated with it. Consequently, use of the name refers not only to the related device but also to the account on the device. The system treats device names in the following manner:

1. Checks the list of user logical names first. Each job can have up to three assigned logical names. (If the logical name has a $ character in it, this step is omitted.)
2. Scans the table of disk logical names for a matching pack identification or equivalent.

3. Checks the additional names.
4. Checks the list of valid physical device names.

If individual users have not defined user logical names that duplicate currently defined system logical names, all users have access to devices by the logical names the system manager adds.

A system logical name must be unique. A pack identification or its logical name cannot conflict with any other pack identification or system logical name. However, you can define multiple system logical names for the same device.

If the pack identification (or a logical name to replace the pack identification) of a disk you are to mount is the same as an existing system logical name, then RSTS/E mounts the disk with no system logical name. Unless you specify a different, unique name, the disk, at mount time, has no logical name associated with the device and unit being used.

System logical names apply for a single time-sharing session; however, you can remove or change them during the session. DIGITAL recommends that you place the proper commands in START.COM to define names automatically at system startup time. See the *RSTS/E Programming Manual* for more details on system logical names.

12.3.1 Assigning Logical Names

The ASSIGN/SYSTEM dev:[PPN] logical-name command defines a system logical name for nondisk devices or an additional name for a disk device. You need INSTAL privilege to use ASSIGN/SYSTEM. If the name duplicates one currently defined and you specify the /REPLACE=QUERY qualifier or you omit the /[NO]REPLACE qualifier, DCL prints a warning message and prompts to see if you want to replace the logical. If you specify /NOREPLACE and the logical name is already defined, DCL prints the error message:

```
?Logical already installed
```

To change a currently defined name, use ASSIGN/SYSTEM/REPLACE command.

The logical name must contain from 1 to 9 alphabetic or numeric characters. You can include the dollar sign ($) character, however, it must not be the first character. If the name contains illegal characters, DCL prints one of the the following error messages:

```
?Invalid logical name
?Unexpected character
```

pointing to the first nonalphanumeric character. If there is no more room in XBUF, the system prints:

```
?No buffer space available
```

The device with which the name is associated must exist on the system. If it does not, the system prints:

```
?Not a valid device
```

12.3.2 Removing Logical Names

The DEASSIGN/SYSTEM command deletes the association defined for the logical name specified. You need INSTAL privilege to use DEASSIGN/SYSTEM. Because a system logical name must be unique, you need to specify only the name in the command. If the name you include is not currently defined, the system prints:

```
?Logical not found
```

If you use an illegally formed name, the system prints one of the following:

```
?Invalid logical name
?Unexpected character
```

pointing to the first nonalphanumeric character. To delete all logical names for a device, you must use the DEASSIGN/SYSTEM command for each currently defined name.

12.3.3 Listing System Logical Names

The SHOW LOGICAL/SYSTEM command displays, for the specified logical name, the device designation and unit number, any account number associated with the logical name, and the logical name itself. If you do not specify a logical name, RSTS/E lists all the system logical names. If you specify the first few characters of a system logical name, RSTS/E lists all the names beginning with those letters. RSTS/E displays the entries in the order in which they are found in the monitor tables.

12.4 Defining Concise Command Language Commands

Concise Command Language (CCL) commands on RSTS/E let users type system-level commands that load and run programs from disk. To take advantage of this feature, you must code your programs to recognize a CCL entry and to extract any command string passed to them. The *RSTS/E Programming Manual* describes the operation and interpretation of CCL commands.

Certain RSTS/E programs can interpret a standard set of CCL commands. A RSTS/E installation can have any number of unique CCL commands. You make all CCL commands available to users by the DEFINE/COMMAND/SYSTEM command which creates the monitor structure required. You need INSTAL privilege to use the DEFINE/COMMAND/SYSTEM command.

Because the monitor structure that defines a CCL command is temporary, CCL commands must be defined at the start of each time-sharing session. It is suggested that you place the proper command definitions in START.COM.

Each CCL command definition occupies one small buffer on the system. All definitions are in a linked list of small buffers. The system executes a SYS call to add or remove CCL definitions and provides a way to list all currently defined CCL commands.

12.4.1 Adding a CCL Definition

To add SY-STAT as a CCL command, execute the DEFINE/COMMAND/SYSTEM command as follows:

```
$ DEFINE/COMMAND/SYSTEM SY-STAT [1,2]SYSTAT.*
```

This command adds the definition for the SYSTAT system program. The hyphen (-) designates the abbreviation point so typing SY runs the SYSTAT program. If you do not include a hyphen in your CCL definition, the system places a hyphen at the end of the command and allows no abbreviation. RSTS/E assumes the public disk structure (SY:) when you do not specify a device. Programs you run with a CCL command can reside on any disk device, but if it is not in the public structure, you must include the device in the command definition. The account designation [1,2] in the example means SYSTAT must reside in the system library. If you do not specify an account with the CCL command, the system assumes the program is located in the account in which the job (that is, the job o the user later invoking the CCL) is running.

The asterisk (*) in SYSTAT.* indicates a wildcard file type. In the case of multipl versions of SYSTAT, the version of SYSTAT that is run depends on the order of the run-time systems in your run-time system list.

You can specify the line number at which execution starts by appending the /LINE_NUMBER=n qualifier to the command. The value of n can be any integer from 0 to 32767. There are standard entry points that you can specify by the use of keywords:

- CCL—30000
- DCL—30500
- CHAIN—31000

The system performs error checking before adding the CCL command. If the command is invalid, RSTS/E prints one of the following error messages:

```
?Invalid command
?Unexpected character
```

pointing to the first nonalphanumeric character. A valid command can consist of:

- A string of 1 to 9 alphanumeric characters
- The first character must be a letter, or one of the following single characters: @ or #

Lowercase letters are equivalent to uppercase in the definition and use of CCL commands. The $ CCL is reserved; it always points to DCL. The command can have a maximum of nine characters. If more than nine characters are present, the system prints:

```
?Command too long
```

When you add two or more commands that begin with the same character or set of characters, you must define the command with the largest number of characters (to the left of the hyphen) first. You then define the command with the next fewest characters to the left of the hyphen, until you have made all the assignments. For example, you must define MACR-O before adding MAC-, or RSTS/E prints:

```
?Command not unique
```

This message also appears if you attempt to define a command that was already defined.

If you need to add a longer CCL definition after you have already added a similar but shorter one, you must first remove the shorter entry. You then add the longer CCL command which lets you reinstall the CCL with the shorter command name. For example, if you had installed MAC- and at some later date needed to add MACR-O, you would have to remove MAC-, install MACR-O, and then reinstall MAC-. The section "Removing a CCL Definition" describes how to remove CCL entries.

12.4.2 Listing Currently Defined CCL Commands

The SHOW COMMAND/SYSTEM command prints a listing of all currently defined CCL commands. You can also list a single CCL command with the SHOW/COMMAND/SYSTEM <name> command. You need INSTAL privilege to see the program name and characteristics in the SHOW COMMAND/SYSTEM command. The list of CCL commands on your system may contain some of the following:

```
$ SHOW COMMAND/SYSTEM
ATT-ACH    = SY:[  1,2  ]LOGIN .*    /LINE=CCL   /PRIVILEGE
BYE-       = SY:[  1,2  ]LOGOUT.*    /LINE=0     /PRIVILEGE
DIR-ECTORY = SY:[  1,2  ]DIRECT.*    /LINE=CCL   /PRIVILEGE
DIS-MOUNT  = SY:[  1,2  ]UMOUNT.*    /LINE=CCL   /PRIVILEGE
EDT-       = SY:[  1,2  ]EDT   .TSK /LINE=0
HELL-O     = SY:[  1,2  ]LOGIN .*    /LINE=0     /PRIVILEGE
HE-LP      = SY:[  1,2  ]HELP  .*    /LINE=CCL   /PRIVILEGE
LBR-       = SY:[  1,2  ]LBR   .TSK /LINE=0
LIBR-      = SY:[  1,2  ]LIBR  .SAV /LINE=8208
LIN-K      = SY:[  1,2  ]LINK  .SAV /LINE=8208
MACR-O     = SY:[  1,2  ]MACRO .SAV /LINE=8216
MAC-       = SY:[  1,2  ]MAC   .TSK /LINE=0
MOU-NT     = SY:[  1,2  ]UMOUNT.*    /LINE=CCL   /PRIVILEGE
PIP-       = SY:[  1,2  ]PIP   .SAV /LINE=8208
PL-EASE    = SY:[  1,2  ]PLEASE.*    /LINE=CCL   /PRIVILEGE
QU-EUE     = SY:[  1,2  ]QUE   .*    /LINE=CCL   /PRIVILEGE
SUB-MIT    = SY:[  1,2  ]QUE   .*    /LINE=CCL   /PRIVILEGE
SY-STAT    = SY:[  1,2  ]SYSTAT.*    /LINE=CCL   /PRIVILEGE
TE-CO      = SY:[  1,2  ]TECO  .TEC /LINE=3584
TKB-       = SY:[  1,2  ]TKB   .TSK /LINE=0
$
```

The system prints the commands in the order in which they were defined. If no CCL commands are currently defined, the system prints the message:

```
%No Commands installed
```

NOTE

When you start the system, the monitor automatically installs the $ CCL. You cannot redefine the $ CCL.

The $ CCL lets you issue DCL commands from any command environment. For example, from BASIC-PLUS you can type:

```
Ready
$ MOUNT DL0:MINDY/NOSHARE <RET>
Ready
```

12.4.3 Removing a CCL Definition

To remove a CCL definition, type the command DELETE/COMMAND/SYSTEM followed by the definition. You need INSTAL privilege to execute the DELETE/COMMAND/SYSTEM command. You can type either a full definition oı a valid abbreviation to indicate the command you plan to remove:

```
$ DELETE/COMMAND/SYSTEM MYPR-OG
```

If the definition does not currently exist, the system prints:

```
?Command not found
```

If the definition exists, the command removes the CCL definition immediately. You can delete all the CCL commands by appending the /ALL qualifier to the DELETE/COMMAND/SYSTEM command. For example:

```
$ DELETE/COMMAND/SYSTEM/ALL
```

12.5 System File Control

You allocate and position the following system files on file-structured disks by executing the INSTALL/SWAP_FILE, INSTALL/OVERLAY_FILE, or INSTALL/ERROR_FILE commands:

- SWAP0.SYS
- SWAP1.SYS
- SWAP3.SYS
- OVR.SYS
- ERR.SYS

These optional files optimize system performance by taking advantage of higher speed disks and the characteristics of user jobs running on the system. Because the files are contiguous and contiguous space becomes scarce as user files are created during timesharing, preallocation and prepositioning are necessary. The *RSTS/E Installation and Update Guide* describes the planning considerations for preallocating and positioning these system files.

RSTS/E does not automatically access these system files. You control RSTS/E access to these files with DCL commands. Thus, the use of these files during timesharing is dynamic. If a hardware problem occurs on a swap device, you can remove the device without stopping timesharing. To maintain the same amount of swap space, you can designate a private disk as the swap device.

This section discusses the DCL commands that control system file usage. The commands in the startup control file allow access to system files at the start of time-sharing operations.

12.5.1 Adding and Removing Swap Files

The swap files SWAP0.SYS, SWAP1.SYS and SWAP3.SYS in account [0,1] are standardized names for the three optional swap files. While these names are optional, Digital recommends you use them because they indicate the characteristics of the files. You can, however, assign any file name to these three swap files. The system restricts the file type to .SYS regardless of the file names you include in the DCL commands.

The INSTALL/SWAP_FILE command causes the system to access a specific file (if a file-structured device is involved) or a specific device (if a non-file-structured disk is involved). You need INSTAL privilege to execute the INSTALL/SWAP_FILE command. A typical command sequence is:

```
$  INSTALL/SWAP_FILE=0 DR3:SWAP0
$  INSTALL/SWAP_FILE=1 DR1:SWAP1
$  INSTALL/SWAP_FILE=3 DR4:SWAP3
```

NOTE

> When you specify a non-file-structured device in the INSTALL/SWAP_FILE command, the monitor destroys any existing file structure on that device.

The /SIZE=n qualifier of the INSTALL/SWAP_FILE command allows creation of a swap file during timesharing. The qualifier is most useful when you want to add a swap file to a private disk that contains sufficient contiguous space for the file. The value n in the /SIZE=n qualifier represents the number of blocks in the file. RSTS/E attempts to access the file with a file type of .SYS in account [0,1] on the associated device. If the file already exists and the file size differs from the size specified in the /SIZE=n qualifier, RSTS/E prints a warning message in the format:

```
%Swap file name already exists a size x blocks
```

The value x is the actual size of the swap file.

If the file does not exist, the system attempts to create a contiguous file of the size specified. If there is not enough contiguous space available to create the file, the system prints:

```
?Not enough contiguous space to create Swap file
```

The /POSITION=n qualifier lets you place the created swap file in a particular location on the specified disk. The monitor ignores this qualifier if the already exists.

After you type the INSTALL/SWAP_FILE command, the system tries to install the particular file or device as the swap file. If the system finds any errors, RSTS/E prints an error message. Appendix D contains a list of the possible errors. If no errors occur, the system installs the swap file.

You can then activate the added swap file with the SET SYSTEM/LOGINS command. The SET SYSTEM/LOGINS command forces the system to examine all swap file space and to calculate the number of jobs the system is capable of running.

During timesharing, the number of jobs that can run on the system depends on:

- The job maximum (JOB MAX) set at system generation
- The amount of swap space added
- The number of logins allowed through the SET SYSTEM/LOGINS command

At the start of timesharing, the number of logins allowed is one. One job is necessary for the DCL START.COM file to run. You can increase this number by using the SET SYSTEM/LOGINS command, but the number is restricted by the swap space available and by JOB MAX. When you add swap space with INSTALL/SWAP_FILE commands, the capacity to handle logins is increased. The SET SYSTEM/LOGINS command sets the number of logins to the maximum allowed. This maximum cannot exceed JOB MAX.

RSTS/E lets you dynamically remove as well as add swap files 0, 1, and 3. To remove a swap file or device, first decrease the number of logins using the SET SYSTEM/NOLOGINS or SET SYSTEM/LOGINS n command. The number of logins to be decreased depends on the capacity of the swap file to be removed. By decreasing the number of logins, you allow the system to safely remove swap space.

The sequence of commands to remove a swap file is the reverse of the sequence to add swap space. For example, assume that a system is running with a JOB MAX of 63 and is using a swap file with a capacity of 16 jobs. In this case, you must decrease the number of logins by 16 jobs before removing the swapfile. The following sequence shows this procedure:

```
$  SET SYSTEM/LOGINS 47
$  REMOVE/SWAP_FILE=1
$
```

In practice, it may take some time for the system to adjust operations. If the number of jobs currently running is greater than 47 or if the number of logins allowed is too large, the system prints the error message:

```
?Too many logins allowed to remove Swap file
```

You must wait until users log out of the system. The SHOW BUFFERS command or the DISPLY program prints the number of jobs currently on the system in the free buffer status report. (The SHUTUP program automatically removes swap files.)

It is possible to add more swap space than the system can use. In this case, you do not need to reduce logins to remove a swap file. However, enough room must remain to swap all jobs currently logged in, or to swap all jobs that can log in, whichever is greater.

The SHOW FILE/SYSTEM command provides a way for you to verify that the following files exist on your system:

- Swap files (/SWAP_FILE)
- Overlay file (/OVERLAY_FILE)
- Error file (/ERROR_FILE)
- DECnet/E Network Services Protocol (NSP) system file (/NETWORK_FILE)

This command is most useful when you want to verify the addition or removal of any of these files. After adding a file, you can use the SHOW FILE/SYSTEM/SWAP_FILE command to verify, for example, that you gave the file the correct file specification, or that the file was added properly. If you add a swap file, such as DR3:[0,1]SWAP0.SYS, execute the SHOW FILE/SYSTEM/SWAP_FILE command as follows:

```
$  SHOW FILE/SYSTEM/SWAP_FILE

Swapfile 0: DR3:[0,1]SWAP0 .SYS 10240 blks (40 jobs at 64K Swap max)
Swapfile 1: None
Swapfile 2: DR1:[0,1]SWAP  .SYS   256 blks (1 job at 64K Swap max)
Swapfile 3: DR4:[0,1]SWAP3 .SYS  1000 blks (3 jobs at 64K Swap max)

$
```

RSTS/E prints None beside the files that do not exist. Note that the system cannot add or remove the DECnet/E file but can only verify that the file exists on your system. There is no DCL command to add or remove this file. Refer to the SET SYSTEM and CLEAR SYSTEM commands in the *DECnet/E System Manager's Guide* for further information.

12.5.2 Adding and Removing Overlay and Error Files

You can add and remove the overlay and error message files during timesharing. OVR and ERR are standardized names to denote the separate file for the overlay code and the alternate file for the default error message file. Although you can use other file names, Digital recommends that you keep the standardized ones. Regardless of the names you give the files, the system requires you to use a file type of .SYS. You need INSTAL privilege to use the INSTALL command.

The INSTALL/OVERLAY_FILE command directs the system to use the specified file for the system overlay code. Similarly, the INSTALL/ERROR_FILE command causes the system to use the specified file when it reads error messages. To add these system files, specify the commands as in the following example:

```
$ INSTALL/OVERLAY_FILE DU2:OVR
$ INSTALL/ERROR_FILE DB0:ERR
```

For the overlay file, the system copies the overlay code from the monitor SIL to the file OVR.SYS on DU unit 2. For the error message file, the system copies into ERR.SYS on DB unit 0 the contents of the error message file (ERR.ERR). You must make sure that the disk on which an overlay or error file resides is mounted when you issue this command. As with swap files, you can use the /SIZE and /POSITION qualifiers for overlay and error files.

To install these files, the system makes sure that the files exist and are properly formed. If the system finds any errors, it prints an error message.

Appendix D contains the text of all error messages. If no errors occur, the system has successfully added the file and is accessing it. You do not need to enter any further commands.

You can remove these files at any time with the REMOVE/OVERLAY_FILE and REMOVE/ERROR_FILE commands. Then, the system accesses the original data. Note that the SHUTUP program automatically removes both files.

12.5.3 Using the DUMP/SYSTEM Command

The DUMP/SYSTEM command lets you take an online dump of the current monitor image executing in memory. When you execute the DUMP/SYSTEM command, RSTS/E copies the memory to the crash dump file CRASH.SYS in account [0,1]. You need SYSIO privilege to use the DUMP/SYSTEM command. Use the DUMP/SYSTEM command as follows:

```
$ DUMP/SYSTEM
```

Once the copy operation is complete, you can use the ANALYS program to analyze the contents of the file (see Chapter 10).

The system returns the following error message if crash dump was not enabled:

```
?Can't find file or account
```

This can happen only if sufficient disk space was not available at system startup to allocate the CRASH.SYS file. To correct this condition, free up the necessary amount of disk space, shut down and then restart the system.

If a user who does not have SYSIO privilege attempts to dump memory with DUMP/SYSTEM, the system prints:

```
?SYSIO privilege required
```

Use of the DUMP/SYSTEM command may also return various device dependent errors. For example:

```
?Device hung or write locked
```

12.6 DCL Command Descriptions

The following sections describe the DCL commands for managing system files and logical names.

12.6.1 ASSIGN/SYSTEM Command

The ASSIGN/SYSTEM command assigns a system-wide logical name to the specified device type and unit. If you include an account number, the ASSIGN/SYSTEM command also associates that project-programmer number with the logical name. ASSIGN/SYSTEM requires INSTAL privilege.

Format

ASSIGN/SYSTEM *dev: [p,pn] name[:]*

Command Qualifiers	Defaults
/[NO]REPLACE[=QUERY]	/REPLACE=QUERY

Prompts
Device:
Logical name:

Command Parameters

dev:
Specifies the device type and unit to which you want to assign a system-wide logical name.

[p,pn]
The optional PPN to which you want to assign a system-wide logical name. DCL does not allow any spaces between the device specification and the PPN.

name[:]
The name that you want added to the system logical name table.

Command Qualifiers

/[NO]REPLACE[=QUERY]
Determines what action to take if the logical name already exists. /REPLACE means replace the logical if it already exists. RSTS/E displays an informational message. /NOREPLACE means display an error message and do not replace the logical if it already exists. If neither is specified or if /REPLACE=QUERY is specified, RSTS/E displays a warning message and prompts to see if you want to replace the logical.

12.6.2 DEASSIGN/SYSTEM Command

The DEASSIGN/SYSTEM command removes a specified logical name from the system logical table. It can also remove logical names for devices on your system. Device mnemonic names for system devices have permanent entries in the logical table. So, if you use the DEASSIGN/SYSTEM command to remove a logical name associated with a device, only the logical name goes away, not the device mnemonic. You can then give a new name to that device. DEASSIGN/SYSTEM requires INSTAL privilege.

Format

DEASSIGN/SYSTEM *name[:]*

Command Qualifiers	Defaults
none	none

Prompts

Logical name:

Command Parameters

name[:]

Specifies the logical name to be removed from the system logical table.

12.6.3 DEFINE/COMMAND/SYSTEM Command

The DEFINE/COMMAND/SYSTEM command creates or redefines a systemwide command (CCL).
DEFINE/COMMAND/SYSTEM requires INSTAL privilege.

Format

DEFINE/COMMAND/SYSTEM *command file-spec*

Command Qualifiers	Defaults
/LINE_NUMBER=entry point	none
/[NO]PRIVILEGE	none

Prompts

Command:
Program name:

Command Parameters

command

Specifies the command you want to define.

file-spec

Specifies the file you want to associate with the command.

Command Qualifiers

/LINE_NUMBER=entry point

Defines the entry point in the command program as follows:

- n—Defines the entry point as the specified integer in the range 0 to 32,767
- CCL—Defines the entry point as the CCL entry point (30000)
- CHAIN—Defines the entry point as the CHAIN entry point (31000)
- DCL—Defines the entry point as the DCL entry point (30500)

/[NO]PRIVILEGE

Specifies whether the program needs to retain its temporary privileges (see Chapter 6).

12.6.4 DELETE/COMMAND/SYSTEM Command

The DELETE/COMMAND/SYSTEM command deletes the specified command from the list of defined system-wide commands (CCLs). The command DELETE/COMMAND/SYSTEM requires INSTAL privilege.

Format

DELETE/COMMAND/SYSTEM

Command Qualifiers	Defaults
/ALL	none

Prompts

Command:

Command Parameters

command

Specifies the command to be deleted from the list of defined system-wide commands.

Command Qualifiers

/ALL

Deletes all system-wide commands.

12.6.5 DUMP/SYSTEM Command

The DUMP/SYSTEM command records in the CRASH.SYS file the current state of the system so you or field service has a way to investigate reasons for system malfunctions. DUMP/SYSTEM requires SYSIO privilege.

Format

DUMP/SYSTEM

Command Qualifiers	Defaults
none	none

Prompts

none

12.6.6 INSTALL/ERROR_FILE Command

The INSTALL/ERROR_FILE command installs the specified file as the system error message file. INSTALL/ERROR_FILE requires INSTAL privilege.

Format

INSTALL/ERROR_FILE *[file-spec]*

Command Qualifiers	Defaults
/[NO]LOG	/LOG
/POSITION=dcn	none
/SIZE=n	none

Prompts

Error file:

Command Parameters

file-spec

Specifies the file to be installed as the system error message file. If you do not specify a file name, the system uses the default file _SY:[0,1]ERR.SYS.

Command Qualifiers

/[NO]LOG

Indicates whether the system displays a confirmation message after it installs the error file. The default is /LOG.

/POSITION=dcn

Places the error file at the specified position on the disk. You must also specify the /SIZE=n qualifier. The value of position can be one of the following:

Value	Meaning
dcn	Locate the file at the first available position on the device past the specified Device Cluster Number.
MIDDLE	Locate the file at the first available position past the middle of the device.
INDEX	Locate the file at the first available position past the Master file Directory (MFD) index.

/SIZE=n

Creates a system file with a size of n blocks. The /SIZE=n qualifier is required if the file does not already exist.

12.6.7 INSTALL/LIBRARY Command

The INSTALL/LIBRARY command adds the named library to the list of installec resident libraries. INSTALL/LIBRARY requires INSTAL privilege.

Format

INSTALL/LIBRARY *name*

Command Qualifiers	Defaults
/[NO]ADDRESS[=n]	See discussion
/[NO]ERROR_LOGGING	/ERROR_LOGGING
/[NO]LOCK	/NOLOCK
/PROTECTION=n	/PROTECTION=42
/[NO]READ_ONLY	/READ_ONLY
/[NO]SHAREABLE	/SHAREABLE
/[NO]UNLOAD	/NOUNLOAD

Prompts

Library:

Command Parameters

name

Specifies the name of the library to add to the list of installed resident libraries.

Command Qualifiers

/[NO]ADDRESS[=n]

Indicates that the resident library should be loaded into memory in 1K-word sections beginning at address n. Specifies the address in memory to load the resident library. Values can be in the range 1K to 2043K.

If n is omitted, RSTS/E computes the starting address by finding the highest memory segment large enough to hold the named library. If this qualifier is omitted, RSTS/E computes a new starting address each time it loads the named library into memory.

/[NO]ERROR_LOGGING

Indicates whether errors that occur within the resident library should be recordec in the system error log. The /ERROR_LOGGING qualifier logs any errors that occur in the resident library to the system error log. The /NOERROR_LOGGING qualifier prevents errors that occur within the resident library from being recorded in the system error log. The default is /ERROR_LOGGING.

/[NO]LOCK

Indicates whether the resident library should remain in memory. The /LOCK qualifier locks the resident library in memory. The /NOLOCK qualifier unlocks the resident library from memory. The default is /NOLOCK.

/PROTECTION=n

Sets the protection code of the resident library. The default code is 42.

/[NO]READ_ONLY

Indicates the allowable type of access to the resident library. The /READ_ONLY qualifier allows read-only access to the resident library. The /NOREAD_ONLY qualifier allows the resident library to be mapped read/write. The default is /READ_ONLY.

/[NO]SHAREABLE

Indicates how many users can access the resident library simultaneously. The /SHAREABLE qualifier lets more than one user access this resident library at the same time. The /NOSHAREABLE qualifier lets only one user access the resident library at any time. The default is /SHAREABLE.

/[NO]UNLOAD

Indicates whether the resident library should remain in memory when no jobs are using it. The /UNLOAD qualifier removes the resident library from memory whenever there are no jobs using it. The /NOUNLOAD qualifier keeps the resident library in memory even when there are no jobs using it. The default is /NOUNLOAD.

12.6.8 INSTALL/OVERLAY_FILE Command

The INSTALL/OVERLAY_FILE command installs the specified file as the system overlay file. INSTALL/OVERLAY_FILE requires INSTAL privilege.

Format

INSTALL/OVERLAY_FILE *[file-spec]*

Command Qualifiers	**Defaults**
/[NO]LOG	/LOG
/POSITION=dcn	none
/SIZE=n	none

Prompts

Overlay file:

Command Parameters

file-spec

Specifies the file to be installed as the system overlay file. If you do not specify a file name, the system uses the default file, _SY:[0,1]OVR.SYS.

Command Qualifiers

/[NO]LOG

Indicates whether the system displays a confirmation message after it installs the overlay file. The default is /LOG.

/POSITION=dcn

Places the overlay file at the specified position on the disk. The /SIZE=n qualifier is required. The value of position can be one of the following:

Value	**Meaning**
dcn	Locate the file at the first available position on the device past the specified Device Cluster Number.
MIDDLE	Locate the file at the first available position past the middle of the device.
INDEX	Locate the file at the first available position past the Master file Directory (MFD) index.

/SIZE=n

Creates a system file with a size of n blocks.

12.6.9 INSTALL/RUNTIME_SYSTEM Command

The INSTALL/RUNTIME_SYSTEM command adds the named run-time system to the list of installed run-time systems. INSTALL/RUNTIME_SYSTEM requires INSTAL privilege.

Format

INSTALL/RUNTIME_SYSTEM *name*

Command Qualifiers	Defaults
/[NO]ADDRESS[=n]	See discussion
/[NO]ERROR_LOGGING	/ERROR_LOGGING
/[NO]KEYBOARD_MONITOR	/NOKEYBOARD_MONITOR
/[NO]LOCK	/NOLOCK
/MAXIMUM_SIZE=s	none
/MINIMUM_SIZE=s	none
/POSITION=n	at end of list
/PREFIX[=x]	/PREFIX=255
/[NO]READ_ONLY	/READ_ONLY
/[NO]SHAREABLE	/SHAREABLE
/TYPE=filetype	See discussion
/[NO]UNLOAD	/NOUNLOAD

Prompts

Run-time system:

Command Parameters

name

Specifies the name of the run-time system to be added to the list of installed run-time systems.

Command Qualifiers

/[NO]ADDRESS[=n]

Specifies the address in memory to load the run-time system. The value of n can be in the range 1K to 2043K. If n is omitted, RSTS/E computes the starting address by finding the highest memory segment large enough to hold the run-time system. If the qualifier is omitted, RSTS/E computes a new starting address each time it loads the run-time system into memory.

/[NO]ERROR_LOGGING

Indicates whether the run-time system should log errors occurring under its control to the system error file. The /ERROR_LOGGING qualifier indicates that this run-time system should log errors. The /NOERROR_LOGGING qualifier indicates that it should not log errors. The default is /ERROR_LOGGING.

/[NO]KEYBOARD_MONITOR

Indicates whether the run-time system has a keyboard monitor. The /KEYBOARD_MONITOR qualifier indicates that this run-time system is a keyboard monitor. The /NOKEYBOARD_MONITOR qualifier indicates that this run-time system is not a keyboard monitor. The default is /NOKEYBOARD_MONITOR.

/[NO]LOCK

Indicates whether the run-time system should reside permanently in memory. The /LOCK qualifier indicates that the run-time system should reside permanently in memory. The /NOLOCK qualifier indicates that the run-time system should not reside permanently in memory. The default is /NOLOCK.

/MAXIMUM_SIZE=s

Uses s in K words as the maximum job size.

/MINIMUM_SIZE=s

Uses s in K words as the minimum size job allowed.

/POSITION=n

Places the run-time system block in the nth position after the default keyboard monitor in the linked list. By default, RSTS/E positions the run-time system at the end of the list. N can be any number in the range from 0 to 255.

/PREFIX[=x]

Causes the run-time system to use the RSTS/E special prefix EMT feature. The value x can be any number from 0 to 255. RSTS/E selects the code 255 by defaul if you do not specify a value for x.

/[NO]READ_ONLY

Indicates whether the run-time system should allow read-only memory access. The /READ_ONLY qualifier indicates read-only memory access. The /NOREAD_ONLY qualifier indicates read/write memory access. The default is /READ_ONLY.

/[NO]SHAREABLE

Indicates whether the run-time system supports multiple users. The /SHAREABLE qualifier indicates that the run-time system supports more than one user at the same time. The /NOSHAREABLE qualifier indicates that the run-time system supports only one job at a time. The default is /SHAREABLE.

/TYPE=filetype

Specifies the default file type for files executed under the run-time system. By default, DCL uses the file type defined in the run-time system file.

/[NO]UNLOAD

Specifies whether RSTS/E unloads the run-time system from memory when no one is using it. The /UNLOAD qualifier removes the run-time system from memory when all users are using another run-time system or when no jobs are using the run-time system. The /NOUNLOAD qualifier keeps the run-time system in memory even when no jobs are using it. The default is /NOUNLOAD.

12.6.10 INSTALL/SWAP_FILE Command

The INSTALL/SWAP_FILE command adds the specified file to the list of installed swap files. Valid values of swap-index are 0, 1, or 3. INSTALL/SWAP_FILE requires INSTAL privilege.

Format

INSTALL/SWAP_FILE *[=]swap-index [file-spec]*

Command Qualifiers	Defaults
/DEVICE=disk-name[:]	none
/[NO]LOG	/LOG
/POSITION=dcn	none
/SIZE=n	none

Prompts

Swapfile number:
Swapfile name:

Command Parameters

swap-index
Specifies the index of the specified file. Valid values are 0, 1, and 3. If you do not enter a file spec, the system uses the default file, _SY:[0,1]SWAPn.SYS, where n is the index number.

file-spec
Specifies the file to add to the list of installed swap files. You must specify the disk.

Command Qualifiers

/DEVICE=disk-name[:]
Specifies a disk to be used as a swapping device. RSTS/E uses this disk as a non-file-structured device; any data on the disk is lost.

If you include the /DEVICE qualifier, you cannot include a file-spec parameter or include the /SIZE or /POSITION qualifiers.

/[NO]LOG
Indicates whether the system displays a confirmation message after it installs the swap file. The default is /LOG.

/POSITION=dcn

Places the created system file at the specified position on the disk. You must also specify the /SIZE=n qualifier. The value of position can be one of the following:

Value	Meaning
dcn	Locate the file at the first available position on the device past the specified Device Cluster Number.
MIDDLE	Locate the file at the first available position past the middle of the device.
INDEX	Locate the file at the first available position past the Master file Directory (MFD) index.

/SIZE=n

Creates a system file with a size of n blocks. The /SIZE=n qualifier is required if the file does not already exist.

12.6.11 REMOVE/ERROR_FILE Command

The REMOVE/ERROR_FILE command removes the system error message file. REMOVE/ERROR_FILE requires INSTAL privilege.

Format

REMOVE/ERROR_FILE

Command Qualifiers	Defaults
none	none

Prompts

none

12.6.12 REMOVE/LIBRARY Command

The REMOVE/LIBRARY command removes the named file as a resident library. REMOVE/LIBRARY requires INSTAL privilege.

Format

REMOVE/LIBRARY *name*

Command Qualifiers	Defaults
none	

Prompts

Library:

Command Parameters

name

Specifies the file to remove from the list of resident libraries.

12.6.13 REMOVE/OVERLAY_FILE Command

The REMOVE/OVERLAY_FILE command removes the system overlay file. REMOVE/OVERLAY_FILE requires INSTAL privilege.

Format

REMOVE/OVERLAY_FILE

Command Qualifiers	Defaults
none	none

Prompts

none

12.6.14 REMOVE/RUNTIME_SYSTEM Command

The REMOVE/RUNTIME_SYSTEM command removes the named run-time system as an entry in the run-time system table. REMOVE/RUNTIME_SYSTEM requires INSTAL privilege.

Format

REMOVE/RUNTIME_SYSTEM *name*

Command Qualifiers	**Defaults**
none	

Prompts

Run-time system:

Command Parameters

name

Specifies the name of the run-time system to remove from the list of installed run-time systems.

12.6.15 REMOVE/SWAP_FILE Command

The REMOVE/SWAP_FILE command removes the specified swap file from the list of installed swap files. Valid values of swap-index are 0, 1, or 3. REMOVE/SWAP_FILE requires INSTAL privilege.

Format

REMOVE/SWAP_FILE *[=]swap-index*

Command Qualifiers	Defaults
none	

Prompts

Swapfile number:

Prompts

Command Parameters

swap-index

Specifies the index of the swap file to remove. Valid values are 0, 1, or 3.

12.6.16 SET FILE Command

The SET FILE command changes the characteristics of the specified file(s). SET FILE requires read and write access to the file(s).

Format

SET FILE *file-spec*

Command Qualifiers	Defaults
/[NO]CACHE	none
/NOCONTIGUOUS	none
/[NO]DELETABLE	none
/[NO]LOG	/NOLOG
/[NO]PLACED	none
/PROTECTION=n	none
/RUNTIME_SYSTEM=name	none

Prompts

Files:

Command Parameters

file-spec

Specifies the file whose characteristics you want to change. You can use wildcard characters in the file specification.

Command Qualifiers

/[NO]CACHE[=SEQUENTIAL | RANDOM]

The /CACHE qualifier specifies that the file is automatically cached when open. The type of caching depends on the file's UFD entry and the specified OPEN MODE. If you use /CACHE alone on a file with no UFD setting or MODE specification, the default is random caching. TUNE privilege is required to use this qualifier. The optional arguments are:

- SEQUENTIAL—Causes RSTS/E to mark the file's UFD entry such that, if the file is cached, it is cached sequentially.
- RANDOM—Causes RSTS/E to mark the file's UFD entry so, if the file is cached, it is cached randomly. Note that you can override the random cache UFD setting with a sequential caching MODE specification.

The /NOCACHE qualifier specifies that the file is not automatically cached when open. To cache such a file, you must specify SET CACHE/ALL or specify a MODE value (see the *RSTS/E Programming Manual*).

/NOCONTIGUOUS

Lets you extend a contiguous file by changing it to noncontiguous.

/[NO]DELETABLE

Indicates whether a file can be deleted or renamed during timesharing. In a directory listing, RSTS/E shows NODELETABLE files with the flag P (protected) after their size. You need SYSIO privilege to use this qualifier.

/[NO]LOG

Indicates whether the file specification of each modified file is displayed at the terminal. If the file specification contains any wildcards, the default is /LOG. Otherwise, the default is /NOLOG.

/[NO]PLACED

Indicates whether, after modification, a file is placed at its present position on the disk.

/PROTECTION=n

Sets the protection code of a file. The protection code must be in the range 0 to 255. You can also use the SET PROTECTION command to perform this function. You need SYSIO privilege to set the privilege bit of the protection code.

/RUNTIME_SYSTEM=name

Associates a run-time system with the file. The name can be any valid run-time system name from the SHOW RUNTIME_SYSTEM display. When the system tries to execute this file, it uses the named run-time system to load the executable file. You need SYSIO privilege to use this qualifier for file(s) whose protection code is 192 or higher.

12.6.17 SHOW COMMAND/SYSTEM Command

The SHOW COMMAND/SYSTEM command displays information about the specified systemwide command. SHOW COMMAND/SYSTEM requires INSTAL privilege for a full listing. Without the INSTAL privilege, SHOW COMMAND/SYSTEM only displays the command you name.

Format

SHOW COMMAND/SYSTEM *command-name*

Command Qualifiers	Defaults
/ALL	/ALL

Prompts

none

Command Parameters

command-name

Specifies the systemwide command whose information you want displayed. If you do not include this parameter, RSTS/E displays all system commands. If you specify the first few letters of a command, RSTS/E displays all the commands beginning with those leters.

Command Qualifiers

/ALL

Displays information about all the systemwide commands.

12.6.18 SHOW FILE/SYSTEM Command

The SHOW FILE/SYSTEM command displays information about the specified system file(s). The default is /ALL. SHOW/FILE/SYSTEM does not require any privilege.

Format

SHOW FILE/SYSTEM

Command Qualifiers	Defaults
/ALL	/ALL
/ERROR_FILE	none
/NETWORK_FILE	none
/OVERLAY_FILE	none
/SWAP_FILE[=swap-index]	all swap files

Prompts

none

Command Qualifiers

/ALL
Displays information about all installed files.

/ERROR_FILE
Displays information about the system error message file.

/NETWORK_FILE
Displays information about the network database file.

/OVERLAY_FILE
Displays information about the system overlay file.

/SWAP_FILE[=swap-index]
Displays information about the specified swap file, where swap-index is 0, 1, or 3. If no argument is given, RSTS/E lists all swap files.

12.6.19 SHOW LIBRARY Command

The SHOW LIBRARY command displays information about all installed libraries. SHOW LIBRARY does not require any privilege.

The following is an example of the SHOW LIBRARY display:

```
$ SHOW LIBRARY

Resident Libraries:
 Name   Prot        Acct       Size  Users  Comments)
HISTOG < 42>       [  1,2  ]   36K     0    Perm, Addr:1628
CSPLIB < 42>  DR1:[  0,1  ]    8K     8    Temp, Addr:117
RMSRES < 42>  DR1:[  0,10 ]    4K     1    Temp, Addr:1624
RMSLBB < 42>  DR1:[  0,10 ]    3K     1    Temp, Addr:1621
RMSLBA < 42>  DR1:[  0,10 ]    4K     0    Temp, Addr:1617
RMSLBD < 42>  DR1:[  0,10 ]    2K     0    Temp, Addr:1615
RMSLBC < 42>  DR1:[  0,10 ]    2K     0    Non-Res, Addr:1613
RMSLBE < 42>  DR1:[  0,10 ]    3K     0    Temp, Addr:1610
RMSLBF < 42>  DR1:[  0,10 ]    4K     0    Temp, Addr:1606
DAPRES < 42>  DR1:[  0,10 ]   10K     0    Temp, Addr:876
EDT    < 42>  DR1:[  0,11 ]   38K     1    Temp, Addr:1568
B2MRES < 42>  DR1:[  0,1  ]   19K     0    Temp, Addr:830

$
```

Format

SHOW LIBRARY

Command Qualifiers	Defaults
none	none

Prompts

none

12.6.20 SHOW LOGICALS Command

The SHOW LOGICALS command displays information about a user-defined logical name. SHOW LOGICALS does not require any privilege.

Format

SHOW LOGICALS *[name]*

Command Qualifiers	**Defaults**
/ALL	/ALL

Prompts

none

Command Parameters

[name]
Specifies the user-defined logical name whose information you want displayed. If you specify the first few letters of the logical name, RSTS/E displays information on all the logical names beginning with those letters.

Command Qualifiers

/ALL
Displays information about all user-defined logical names.

12.6.21 SHOW LOGICAL/SYSTEM Command

The SHOW LOGICAL/SYSTEM command displays information about a systemwide logical name. You need no special privileges to use SHOW LOGICAL/SYSTEM.

Format

SHOW LOGICAL/SYSTEM *name*

Command Qualifiers	**Defaults**
/ALL	/ALL

Prompts

none

Command Parameters

name

Specifies the systemwide logical name whose information you want displayed.

Command Qualifiers

/ALL

Displays information about all systemwide logical names.

12.6.22 SHOW RUNTIME_SYSTEM Command

The SHOW RUNTIME_SYSTEM command displays information about all installed run-time systems. SHOW RUNTIME_SYSTEM does not require any privilege. The following is an example of the display:

```
$ SHOW RUNTIME_SYSTEM

Run-Time Systems:
 Name   Ext  Dev    Size   Users Comments
...RSX  TSK          0(64)K  14   Monitor, KBM
DCL     COM  DR1:   28(4)K    8   Temp, Addr:77, DF KBM
RT11    SAV  DR1:    4(28)K   0   Temp, Addr:109, KBM, CSZ, EMT:255
TECO    TEC  DR1:  10(20)K    0   Temp, Addr:383, KBM
BAS4AL  BAC  DR1:  17(12)K    0   Non-Res, KBM, CSZ
BAS4F   BAC  DR1:  16(16)K    0   Temp, Addr:322, KBM, CSZ
BASIC   BAC  DR1:  15(16)K    1   Temp, Addr:133, KBM, CSZ
FORTH   4TH  DR1:    6(24)K   0   Temp, Addr:125, KBM, NER, CSZ
GMBODT  COM  DR0:   27(4)K    0   Non-Res, Addr:1541, KBM, 1US, R/W
MOMODT  COM  DR0:   27(4)K    1   Temp, Addr:849, KBM, 1US, R/W, NER
```

Format

SHOW RUNTIME_SYSTEM

Command Qualifiers	Defaults
none	none

Prompts

none

12.6.23 UNLOAD/LIBRARY Command

The UNLOAD/LIBRARY command removes the resident library you specify from memory. UNLOAD/LIBRARY requires the INSTAL privilege.

Format

UNLOAD/LIBRARY *name*

Command Qualifiers	**Defaults**
none	

Prompts

Library:

Command Parameters

name

Specifies the name of the resident library to remove from memory.

12.6.24 UNLOAD/RUNTIME_SYSTEM Command

The UNLOAD/RUNTIME_SYSTEM command unloads the named run-time system from memory. UNLOAD/RUNTIME_SYSTEM requires the INSTAL privilege.

Format

UNLOAD/RUNTIME_SYSTEM *name*

Command Qualifiers	**Defaults**
none	

Prompts

Run-time systems:

Command Parameters

name

Specifies the name of the run-time system to remove from memory.

Chapter 13

Job Management

This chapter describes the DCL commands that you use to manage RSTS/E jobs. Table 13–1 summarizes these commands. The rest of this chapter explains the commands in detail and gives examples of their use.

Table 13–1: DCL Commands for Job Management

Command	Description
BROADCAST	Displays a message at the specified destination(s).
DETACH	Detaches a job from its terminal.
FORCE	Forces a "string" to the specified destination(s).
HANGUP	Disconnects the specified remote line.
LOGIN	Creates a job at the specified terminal, logs your current job in under a different account, or resets your current job to its initial logged-in state.
REMOVE/JOB	Immediately terminates the specified user job and logs it off.
SET JOB	Defines the characteristics of a job.
SET SYSTEM	Sets the default system characteristics.
SHOW JOB/PRIVILEGE	Displays the set of current privileges for the current job.

13.1 Controlling the Number of Logged-In Jobs

You can monitor and control system operation while logged in to a RSTS/E system. With the SYSTAT or the DISPLY system program, you can observe how the system is performing. If performance declines, you can remedy the problem, in some cases, by preventing more users from logging in to the system. You adjust the number of logged-in jobs allowed with the SET SYSTEM/LOGINS, and SET SYSTEM/NOLOGINS commands.

SET SYSTEM/NOLOGINS sets to one the number of jobs that are allowed to log in to the system. After RSTS/E executes the SET SYSTEM/NOLOGINS command, the system allows no more users to log in to the system. If a user attempts to log in, the monitor prints the following error message:

```
?Logins are disabled - please try again later
```

Jobs already logged in to the system can continue running. However, jobs such as BATCH that create logged in jobs cannot successfully process further requests. (The SHUTUP system program also disables further logins in preparing to shut down timesharing.)

If you try to connect to RSTS/E over a LAT line and get the message "?Logins are disabled—please try again later", RSTS/E does not disconnect the session. Instead, the terminal waits for logins to become enabled; then you can start the login sequence by pressing Return. If you want to get back to the LAT prompt, use the local key sequence, usually the Break key.

To keep you from being locked out of the system, RSTS/E allows jobs to be logged in at the system console terminal (KB0:), regardless of the number of logins currently allowed. (This feature is a special characteristic of the terminal you designate as the system console; the default is KB0:. By installing a feature patch, you can change the terminal which has this characteristic. However, only one terminal on a system can have this capability.) This capability remains in effect as long as a job slot is available to handle a new job. You need SWCTL privilege to use the SET SYSTEM/NOLOGINS command.

The SET SYSTEM/LOGINS command places a ceiling on the number of users that can log in to the system. This command is useful for limiting or extending the load allowed on the system. For example:

```
$ SET SYSTEM/LOGINS=35
$
```

This command limits the number of logins to 35; if there are 35 or more users logged in to the system, the next user who tries to log in gets the error message "?Logins are disabled—please try again later," and must wait until enough users log off before being allowed to log in.

The system does not let you set the number of logins to zero or to a number greater than the value of JOB MAX set at system installation time. If you specify zero in the command, RSTS/E sets the number to one. The default number is JOB MAX. You need SWCTL privilege to use the SET SYSTEM/LOGINS command.

The number of logins allowed can never exceed the capacity of the swap space installed on the system or the value specified during system generation. If you enter a number greater than the maximum allowed, RSTS/E sets logins to the maximum possible. The SHOW BUFFERS command or the DISPLY program includes the maximum number of logins currently allowed.

You normally use the SET SYSTEM/LOGINS command with the INSTALL/SWAP_FILE command to adjust the number of logins allowed. After you add swap space to the system, the SET SYSTEM/LOGINS command performs these steps:

1. Recalculates the swap capacity
2. Factors in the currently allowed maximum swap size (SWAP MAX)
3. Increases the number of allowed logins

If you are preparing to remove swap space, use the SET SYSTEM/LOGINS command to decrease the number of allowed logins.

13.2 Broadcasting Messages to Terminals

The BROADCAST command lets you communicate with one user or with all users on the system. You need SEND privilege to execute the BROADCAST command. This command places a specified text string in the output buffer of a terminal, or of all terminals, and displays the text on the terminal. For example, if a user at terminal KB32: assigns a peripheral device for an unreasonably long time, you can send a message asking the user to deassign the device:

```
$ BROADCAST KB32: "Bob, can you deassign MM0: for me; I need it ASAP."
$
```

By specifying /LOGGED_IN instead of a terminal designator, you can broadcast the message to each online terminal in the RSTS/E system (unless the terminal is set for NOBROADCAST). For example:

```
$ BROADCAST/LOGGED_IN "Read $NOTICE.TXT; the system is coming down!!"
$
```

The system displays your message as follows:

```
*** From [1,214] "Account Name" on KB25: at 02:17 PM 27-May-85
** Read $NOTICE.TXT; the system is coming down!!
```

This message appears on all terminals that are on line and set to BROADCAST, except your own.

You can direct your message to specific users by substituting a PPN for the destination specification. Wildcard characters are also permissible. For example, to broadcast a message to all users logged in under the same project number:

```
$ BROADCAST [20,*] "Please attend a meeting at 3:00 PM today"
```

You can send a multiline message with BROADCAST as follows:

1. Press Return after the destination specification. RSTS/E displays the prompt Message: on your terminal.
2. Enter your message, terminate each line by pressing Return.
3. Enter Ctrl/Z to send your message.

In a multiline message, you do not need to enclose the message in quotes. For example:

```
$ BROADCAST/ALL<RET>
Message: The power will be shut off in the building this weekend.<RET>
Message: Therefore, this system will be taken down Friday night.<RET>
Message: No logins will be allowed after 5PM on Friday.<RET>
Message: <Ctrl/Z>
$
```

13.3 Controlling Jobs

The FORCE command lets you send a text string to another terminal (or all terminals). You need SYSIO privilege to use the FORCE command. The system places the text string in the input buffer of the specified terminal as if it were typed by the user. The command has the format:

FORCE destination "command"

If you want to send a Ctrl/C to a terminal, KB32 in this case, to stop the execution of whatever is running at the terminal and then to log the user off the system, type:

```
$ FORCE KB32: "^LOGOUT"
$
```

The circumflex (^) character before the command forces a Ctrl/C to the terminal, and the LOGOUT command logs the user off the system.

You can also send only a Ctrl/C (^C) to a terminal by placing a circumflex character in the first position after the keyboard number. For example:

```
$ FORCE KB32: "^"
$
```

Do not place any text after the circumflex if you want to send only a Ctrl/C to the terminal to stop the job that is running. RSTS/E sends a Ctrl/C to the terminal and returns control to the keyboard monitor prompt. For example:

```
^C

$
```

To force a control character combination (Ctrl/Z for example) to a terminal, enter the circumflex (^) character as the first character of the text, followed by the proper control letter (such as Z). For example:

```
$ FORCE KB32: "^Z"
$
```

Typing ^Z and pressing Return executes a Ctrl/Z at KB32: No other text should follow the control character combination. If the circumflex (^) is the first character of more than two characters of text, however, RSTS/E forces a Ctrl/C to the terminal before sending the text that immediately follows the first character. For example:

```
$ FORCE KB32: "^!Harry, sorry but your job is out of control!"
$
```

The ^C characters appear on Harry's terminal to end the job he was running and the system prints the text of the message on the following line. (The text after the first character can contain up to 80 characters.)

13.4 Removing Jobs

The REMOVE/JOB/NOQUERY command lets you terminate a user's job and remove it from the system. You need JOBCTL privilege to use the REMOVE/JOB/NOQUERY command. Before using REMOVE/JOB/NOQUERY, use the SHOW USER command to learn the user's job number, then type:

```
$ REMOVE/JOB/NOQUERY 10
$
```

The monitor clears job 10 and frees it for other system use. The user is no longer logged in to the system and must log in again before attempting any other system activity. Use the SHOW USER command immediately after issuing the REMOVE/JOB/NOQUERY command to verify that the job is no longer on the system.

13.5 Placing a Job on HOLD

There are times when you do not want to terminate a job, but rather want to place it in a hold state, a state from which it can resume normal processing. A job may be using up too many system resources, or you may be running a batch job that you need to correct. The SET JOB/HOLD command automatically sets the priority of a job to -128. This causes the system to stop giving the job any time to run. Use the SET JOB/HOLD command as follows:

```
$ SET JOB/HOLD 10
$
```

Job 10 stops processing but does not terminate. SET JOB/HOLD always sets the priority to -128, which means you cannot include a priority number with the command. You must reset the priority of the job to -120 or higher with the SET JOB/RELEASE command to reactivate the job. Use the SET JOB/RELEASE command as follows when you want to restart the job:

```
$ SET JOB/RELEASE 10
$
```

Now job 10 has the default priority of -8, which under normal system load allows the job to successfully run to completion.

SET JOB/HOLD and SET JOB/RELEASE are the logical commands to use when you want to stop a job temporarily and then start it again. You need TUNE privilege to use the SET JOB/HOLD or the SET JOB/RELEASE commands.

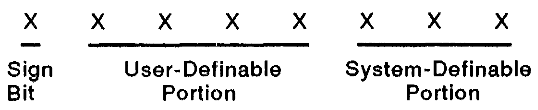
13.6 Setting Job Priority, Run Burst, and Maximum Size

The system lets you set the priority, run burst, and maximum size of an existing job. You can change any of the current values to:

- Increase or decrease the chance of gaining run time in relation to other running jobs
- Determine how much CPU time the job can have when it is compute bound
- Increase the area a job can occupy

The system runs jobs on the basis of priority. The higher a job's priority, the better its chances of obtaining run time in relation to other jobs that are running. Figure 13–1 shows how an 8-bit priority byte determines priority.

Figure 13–1: Priority Byte Format



Using the SET JOB/PRIORITY command, you or another user can set the user-definable portion of the priority byte for any job on the system. Because the three system-definable bits are normally zero, standard priorities are multiples of 8 between -120 (lowest priority) and +120 (highest priority). Zero is a legal priority. When the system changes a priority, it truncates any value not a multiple of eight to the next lowest legal priority. For example, if you specify a priority of +10, RST/E sets the value to +8.

You need TUNE privilege to use the SET JOB/PRIORITY command.

All of the system-definable bits are normally off (zero). The least significant bit is set when a keyboard delimiter is typed and the job is waiting for keyboard input or when any I/O completes. Keyboard delimiters are Carriage Return (CR), Line Feed (LF), Form Feed (FF), Control Z (Ctrl/Z), and Escape Sequence Character (ESC). The next significant bit is set whenever a Ctrl/C is entered and can be set at any time. Finally, a system function call sets the most significant bit of the system-definable portion. RSTS/E clears all system-definable bits when it chains to another program or when the job is at the system command level.

The system-definable portion of the priority byte is always less significant than the user-definable portion. Therefore, the system-definable bits affect priority only within the user-definable priority range. For example, if two jobs are running under priority -8, the user who enters a Ctrl/C has a higher priority (that is, priority -6 in this case) than the user who does not. However, a third user with priority 0 supersedes two users whose priorities are -8 and -6.

When you log in, LOGIN runs with priority 0 and automatically sets your job to priority -8. This is the default priority with which most or all jobs are run. Do not assign priorities other than -8 except in unusual circumstances (for example, compute-bound or detached jobs). On occasion, you may want to run a program that requires a great deal of computation. If time is not a factor in obtaining results, and you have TUNE privilege, you can decrease the job priority of the compute-bound job to improve efficiency for the other users on the system. On the other hand, infrequently used detached programs often have higher priorities (typically priority 0) because they must run quickly when needed but do not run compute bound for an extended period and do not run often.

Run burst is the maximum time a job can run compute bound before another job obtains access to the CPU. On systems using the KW11L line frequency clock, each unit of run-burst time is equal to 1/60th or 1/50th of a second, depending on the system's power line frequency. Systems running with the KW11P clock at crystal speeds, rather than at line frequency, have a run-burst unit of 1/50th of a second. If the system is operating from a 60 Hz power line, one run-burst unit equals 1/60th of a second. In that case, six units equal 1/10th of a second, which is the run-burst default value.

If you assign a run burst of six to a specific job that does not require much compute-bound time, the system automatically transfers control to the next user before the six units have been used. One tenth of a second is generally considered the best run-burst time period to ensure efficient overall system operation. If a job is guaranteed to become I/O bound (that is, I/O stalled) after a certain amount of computation, use SET JOB/RUN_BURST to specify a run burst larger than six. In many cases, a run burst greater than six has a significant effect on long computational programs. Their percentage of the overall CPU time would be increased.

You need TUNE privilege to use the SET JOB/RUN_BURST command.

The maximum size assigned to a job refers to the allowable memory space. You can restrict certain jobs to run only programs smaller than 32K words (or SWAPMAX) by assigning a job a size smaller than the current SWAPMAX. This assigned limit does not affect privileged, compiled programs. Thus, a user with a small amount of space can still run system programs that would normally exceed the memory limit. To alter the maximum job size of job 14, use the SET JOB/SIZE command as in the following example:

```
$ SET JOB/SIZE=16 14
```

You need the TUNE privilege to use the SET JOB/SIZE command.

13.7 Controlling Keyboards and Remote Lines

If you determine that a dataset line is in use but no keyboard activity is taking place (by using SYSTAT or the DISPLY program job status report), you can disconnect the dataset. The HANGUP command disconnects the remote line specified by the KBn: keyboard. The hangup capability prevents a user from monopolizing the line without being charged for connect time and frees the line for other remote users. You need HWCTL privilege to use the HANGUP command.

The DETACH command frees a terminal from its job. If the job does not have the terminal open on a nonzero channel, DETACH frees the terminal for other use. The command normally forces the system to close all nonzero channels on which the terminal is open. The /NOCLOSE switch suppresses this action. When the job specified is currently detached, the monitor prints the error message:

```
?Job number x is already detached.
```

You need JOBCTL privilege to use the DETACH command (except for your current job).

13.8 DCL Command Descriptions

The following sections describe the DCL commands that you can use for job management.

13.8.1 ATTACH Command

The ATTACH command attaches a terminal to a job. This command does not require any privileges.

Format

ATTACH *n*

Command Qualifiers	Defaults
none	none

Prompts

Job number:

Command Parameters

n

The number of the job you want to attach to.

13.8.2 BROADCAST Command

The BROADCAST command sends a message to:

- A specified terminal (KBn:)
- All users logged in under a specified PPN (the PPN can contain wildcard characters)

NOTE

This command displays a message only on terminals that are not set to NOBROADCAST.

You need SEND privilege to use BROADCAST.

Format

BROADCAST *destination ["message"]*

Command Qualifiers	**Defaults**
/ALL	none
/[NO]BELL	/NOBELL
/[NO]HEADER	/HEADER
/LOGGED_IN	/none

Prompts

To:
Message:

Command Parameters

destination
The device(s) or account(s) where the message is to be displayed. For multiple destinations, enter the individual keyboards, separated by commas, for example:

```
$ BROADCAST [4,*],KB23:,KB5 "Please log off."
```

"message"
The message to be displayed on the destination device(s). If you type Return after entering the destination specification, DCL prompts you for each line of a message. Enter Ctrl/Z to send the message. Note that when you enter a multi-line message in this way, do not enclose the message in quotes.

Command Qualifiers

/ALL
Broadcasts the message to all terminals.

/[NO]BELL
Broadcasts a BELL character before the message. The default is /NOBELL.

/[NO]HEADER

Broadcasts a header before the message. The header is in the format:

```
From ARK::[1,194] "Account Name" on KB52: at 11:52 AM 30-May-85
```

The default is /HEADER.

/LOGGED_IN

Broadcasts the message to all logged-in terminals.

13.8.3 DETACH Command

The DETACH command detaches a job from its terminal. You need JOBCTL privilege to detach any job other than your current job.

Format

DETACH *n*

Command Qualifiers	**Defaults**
/[NO]CLOSE	/CLOSE

Prompts
Job number:

Command Parameters

n
The number of the job you want to detach from its terminal.

Command Qualifiers

/[NO]CLOSE
Indicates whether the system closes all nonzero channels on which the job's terminal is open before detaching. The default is /CLOSE.

13.8.4 FORCE Command

The FORCE command forces a string to:

- A specified terminal
- All users logged in under a specified PPN. The PPN can contain wildcards.

You need SYSIO privilege to use FORCE.

Format

FORCE *destination "string"*

Command Qualifiers	Defaults
/ALL	none
/LOGGED_IN	none

Prompts

To:
Command:

Command Parameters

destination
The device(s) that you want the "string" forced to. For multiple destinations, enter the individual devices, separated by commas, for example:

```
$ FORCE [4,*]KB23:,KB5 "^ $LOGOUT"
```

"string"
The string that you want forced to the destination device(s).

Command Qualifiers

/ALL
Forces the string to all terminals.

/LOGGED_IN
Forces the string to all logged-in terminals.

13.8.5 HANGUP Command

The HANGUP command disconnects the specified remote line. You need HWCTL privilege to use HANGUP.

Format

HANGUP *KBn:*

Command Qualifiers	**Defaults**
none	none

Prompts

Keyboard:

Command Parameters

KBn:

Specifies the remote line you want to disconnect.

13.8.6 LOGIN Command

The LOGIN command is used to:

- Create a new job at the specified terminal. You can use this feature in your system startup procedure to log in various users automatically. You need WACNT or GACNT privilege to create a new job. If the terminal is restricted, you also need DEVICE privilege.
- Log your current job in under a different account. This feature lets you switch between accounts without having to first issue the LOGOUT command (see *RSTS/E System User's Guide*). You need WACNT or GACNT privilege to log in to a different account without being prompted for that account's password.
- Reset your current job to its initial logged-in state (the state your job was in after logging in). This feature is useful for resetting logical assignments, DCL symbols, prompts, and so on to their initial state. This feature does not require any privilege.

Format

LOGIN *[p,pn]*

Command Qualifiers	**Defaults**
/[NO]OVERRIDE[=NOLOGINS]	/NOOVERRIDE
/TERMINAL=terminal[:]	Your terminal

Prompts
Password:

Command Parameters

[p,pn]
The PPN for the account you want to log in to. The default PPN is your PPN.

Command Qualifiers

/OVERRIDE[=NOLOGINS]

/NOOVERRIDE
Indicates whether to create the new job if logins are currently disabled. Specify /OVERRIDE or /OVERRIDE=NOLOGINS to create the new job even if logins are disabled. If you attempt to create a new job with the /NOOVERRIDE qualifier and logins are disabled, the monitor displays an error message. The default is /NOOVERRIDE. The /OVERRIDE qualifier requires that you also specify the /TERMINAL qualifier.

NOTE

> Even with /OVERRIDE in effect, you cannot create a new job unless there are job slots available on the system.

/TERMINAL=terminal[:]

Indicates that a new job should be created at the specified terminal. The terminal argument can be any valid device name that refers to a keyboard device. If the terminal you specify is restricted, you must have DEVICE privilege. You must have WACNT or GACNT to log in to another account on the specified terminal.

The following examples show the different features of LOGIN. To reset your job to its initial logged-in state, enter:

```
$ LOGIN
```

To log in to account [10,10], enter:

```
$ LOGIN [10,10]
Password:
```

You must then supply the correct password. If you have WACNT or GACNT privilege and you are switching from any [10,*] account, the monitor lets you log in without prompting you for the password.

To log in user [100,3] at terminal KB11: while you are working at another termingal, enter the following:

```
$ LOGIN/TERMINAL=_KB11: [100,3]
```

You must have WACNT privilege, or have GACNT privilege and be logged in to a [100,*] account to execute the previous command. You must have the DEVICE privilege, or you must have _KB11: set to NORESTRICT.

13.8.7 REMOVE/JOB Command

The REMOVE/JOB command immediately terminates the specified user job and logs it out. You need JOBCTL privilege to use this command.

Format

REMOVE/JOB *n*

Command Qualifiers	Defaults
/[NO]CONFIRM	/CONFIRM
/[NO]QUERY	/QUERY

Prompts

Job number:

Command Parameters

n
Specifies the job to remove.

Command Qualifiers

/[NO]CONFIRM
Causes a SHOW JOB display for the job you want to remove, then asks if you really want to remove the job just displayed.

/[NO]QUERY
Synonym for the /[NO]CONFIRM qualifier.

13.8.8 SET JOB Command

The SET JOB command defines the characteristics of the specified job. The /KEYBOARD_MONITOR and /PRIVILEGE qualifiers can only be used for the current job. If you specify those qualifiers with a job number other than your own, RSTS/E displays the following error message:

```
?Only current job allowed.
```

You need TUNE privilege to use any of the qualifiers except /PRIVILEGE and /KEYBOARD_MONITOR.

Format

SET JOB *[n]*

Command Qualifiers	Defaults
/HOLD	No change
/KEYBOARD_MONITOR[=name]	none
/PRIORITY[=p]	none
/PRIVILEGE=(priv,priv,...)	none
/RELEASE	none
/RUN_BURST[=q]	none
/SIZE=q	none

Prompts	
Job number:	Displayed if you use the /HOLD qualifier

Command Parameters

[n]
Specifies the job whose characteristics you want to define. If you leave off the job number, the default is your own job number.

Command Qualifiers

/HOLD
Sets the priority of job n to -128. This qualifier cannot be used for the current job.

/KEYBOARD_MONITOR[=name]
Sets the current job's keyboard monitor to name. The allowable values of name are any installed run-time system that has a keyboard monitor. Use the SHOW RUNTIME_SYSTEM command to list the run-time systems on your system. If you omit the optional argument, DCL becomes the current job's keyboard monitor.

/PRIORITY[=p]
Sets the priority of the specified job. Priority can be any number from -128 to +120. The system rounds the priority down to a multiple of eight. (For example, if you select a priority of -19, the system sets the priority to -24.) If you do not specify a priority value, the default is -8.

/PRIVILEGE=(priv,priv,...)
Modifies the set of current privileges for the current job only.

/RELEASE

Resumes job n at priority of -8.

/RUN_BURST[=q]

Changes the run burst of the specified job to q. The value for q can be any number from 1 to 127. If you do not specify a run burst value, the default is 6.

/SIZE=q

Changes the maximum size of the specified job to q. You can set q to a maximum of 32K, or 64K for I&D space machines. The system automatically lowers any higher value to the maximum allowed.

13.8.9 SET SYSTEM Command

The SET SYSTEM command sets the default system characteristics. You need SWCTL privilege to use the SET SYSTEM command.

Format

SET SYSTEM

Command Qualifiers	Defaults
/[NO]LOGINS[=n]	See Discussion

Prompts

none

Command Qualifiers

/[NO]LOGINS[=n]

Defines the maximum number of users who can log in to the system at one time. Set /LOGINS=n to the number of user jobs that can log in to the system at one time. The default is JOB MAX. Specify /NOLOGINS if you want to prevent users from logging in to the system by setting the number of logins to one.

See Chapter 14 for additional SET SYSTEM command qualifiers.

13.8.10 SHOW JOB/PRIVILEGE Command

The SHOW JOB/PRIVILEGE command displays the set of current privileges for the specified job. You need JOBCTL privilege to display the privileges of another job.

Format

SHOW JOB/PRIVILEGE *[n]*

Command Qualifiers	Defaults
none	none

Prompts

none

Prompts

Command Parameters

[n]
Specifies the job whose current privileges you want to display. The default is the current job.

Chapter 14

Miscellaneous System Management Functions

This chapter describes miscellaneous DCL commands that you use to manage your RSTS/E system. Table 14–1 summarizes the DCL commands available to you. The rest of this chapter explains each command in more detail.

Table 14–1: DCL Commands for Miscellaneous System Functions

Command	Description
LOAD/INDEX	Loads SAT into memory.
LOAD/OVERLAY	Loads one or more monitor overlays in memory.
SET DATE	Sets the RSTS/E system date to the value you specify. Optionally, you can specify a new value for the system time.
SET SYSTEM	Sets the default system characteristics.
SET TIME	Sets the RSTS/E 24-hour clock to the value you specify. Optionally, you can set the system date to the value you specify.
SHOW SYSTEM	Displays the system-default characteristics.
UNLOAD/INDEX	Unloads SAT from memory.
UNLOAD/OVERLAY	Unloads one or more monitor overlays from memory.

14.1 Loading and Unloading Resident SAT

The LOAD/INDEX and UNLOAD/INDEX commands let you load and unload the disk Storage Allocation Table (SAT) into memory during timesharing. Each disk on the system has a SAT, which is called SATT.SYS.

The SAT is a 1 to 16 block permanent structure on all file-structured RSTS/E disks. Its purpose is to provide the file processor with a table of allocated or unallocated pack clusters, so that the processor can obtain new disk space for file structures. By default, whenever new disk space is allocated on the disk, the system updates the SAT and rewrites it to the disk, showing the true state of in-use disk space.

The LOAD/INDEX and UNLOAD/INDEX commands let you optionally create a SAT that is resident in memory during timesharing. This feature improves system performance because the RSTS monitor does not need to perform disk I/O to the SAT on a per-disk unit basis. By allocating the appropriate amount of XBUF, you can make the SAT for each disk unit completely memory resident, so that a write to the disk occurs only when the disk unit is logically dismounted.

NOTE

Because the resident SAT feature improves system performance at the expense of memory, be sure to calculate the amount of XBUF needed according to the number and size of any resident SATs desired before starting timesharing.

14.2 Loading and Unloading Monitor Overlays

The LOAD/OVERLAY and UNLOAD/OVERLAY commands let you load and unload monitor overlays during timesharing. You can use these commands in conjunction with the DISPLY program and the SHOW CACHE command to monitor the effects on your system's performance of making certain overlays resident or nonresident.

The monitor overlays that you can specify with these commands are:

- ATTRIBUTE—Performs file and account attribute read/write operations
- DCL—Performs file-related operations for DCL
- DELETE_RENAME—Performs file deletion and renaming
- DIRECTORY—Performs disk file lookup operations
- SYSTEM_CALLS—Performs specific monitor directives
- TERMINAL—Performs terminal characteristic SET and SHOW operations

14.3 Suspending System Operations

The SET SYSTEM/HOLD and SET SYSTEM/RELEASE commands let you temporarily suspend all disk I/O and other system operations. When you issue the SET SYSTEM/HOLD command, all jobs on the system are suspended, except for the job issuing the command. Operations resume when you issue the SET SYSTEM/RELEASE command.

The SET SYSTEM/HOLD and SET SYSTEM/RELEASE commands are used primarily on systems where the system disk is an RC25. The main reason for using these commands is to suspend operations when you need to stop the fixed disk drive so you can mount a different disk cartridge (both disks share the same spindle). With SET SYSTEM/HOLD and SET SYSTEM/RELEASE, a total system shutdown is not necessary. You need HWCTL privilege to use the SET SYSTEM/HOLD and SET SYSTEM/RELEASE commands. To use the commands, type:

```
$ SET SYSTEM/HOLD
System now on hold
$
```

The system displays the message System now on hold to show that the suspend is in effect. When the disk stops, you can change the removable cartridge. When you are ready to resume system operations, type:

```
$ SET SYSTEM/RELEASE
System released

$
```

14.4 Changing System Date and Time

The SET DATE and SET TIME commands let you change the system date and the 24-hour clock. These commands are important on systems where strict accounting data is kept and incorrect values may have been entered at system startup time. You can enter the date in either of two formats:

```
SET DATE dd-mmm-yy
SET DATE yy.mm.dd
```

The following example shows how to enter the date using the first format:

```
$ SET DATE 30-MAY-85
```

You can enter the same date in the second format as well:

```
$ SET DATE 85.05.30
```

Be sure to place a space between the DATE command and the beginning of the date field when entering a date with either format. You need DATES privilege to use the SET DATE or SET TIME command.

You can enter the time in either of two formats:

```
SET TIME 01:30PM
SET TIME 13:30
```

You change the date and/or the time with either command as follows:

For SET DATE;

```
$ SET DATE 30-MAY-85
$ SET DATE 30-MAY-85:01:30PM
```

For SET TIME;

```
$ SET TIME 01:30PM
$ SET TIME 30-MAY-85:01:30PM
```

Note that when you specify both parameters, you must first enter the date and separate the two parameters with a colon.

You can also specify a relative date and time with the SET DATE command. For example:

```
SET DATE -1DAY
SET TIME +1HOUR
```

The first command, when executed, sets the system date back a day. The second command, when executed, adjust the system time ahead one hour. You can use relative values to make minor adjustments in the system date or time. Relative date and time syntax is as follows:

+nnDAYS
+nnHOURS
+nnMINUTES

You can string the keywords together in any order and you can abbreviate them to one character. If you specify an absolute date and time, it must appear first in the date-time specification. An example of mixed absolute and relative date and time would be:

```
SET DATE 10-MAY-85:4:00PM+3DAYS+2HOURS-45MINUTES
```

This would be equivalent to:

```
SET DATE 13-MAY-85:5:15PM
```

14.5 Recording System Activities: EMT Logging

An EMT is a PDP-11 assembly language instruction through which a program requests services from the monitor (for example, opening and closing a file, performing I/O, and logging in or out.) EMT logging is an optional feature that provides a "window" on the process by which timesharing jobs request and receive services from the RSTS/E monitor. Thus, EMT logging lets you gather information about the activity on your system.

For example, you might want to know the following information about your system:

- The number of logins on a particular terminal
- How many files are accessed on a certain drive
- Which nonresident FIP overlays get the heaviest use

Such information can help you improve system performance, identify bottlenecks, establish fee setting algorithms, and watch for potential security problems.

This section describes the EMT logging feature, how it works, and how to include it in your system. However, because of the wide variety of RSTS/E systems, you must decide which data is most useful in your environment, and how to collect and use the data the EMT logger provides.

EMT logging includes optional code in the monitor, uses XBUF to pass information, and requires an EMT logging program (which you must supply; a sample is provided in UNSUPP$:), running as a timesharing job. The use of EMT logging can affect performance. This effect depends on which EMTs you decide to log, for which jobs you log them, and how much processing your logging program does for each EMT.

EMT logging provides information on timesharing activity in terms of what the monitor sees. The data returned to your logging program is in terms of FIRQB and XRB contents, regardless of the programming environment of the job that issues the directives. See the *RSTS/E System Directives Manual* for information on the FIRQB and XRB, as well as descriptions of the MACRO form of system directives.

14.5.1 Programming for the EMT Logger

To use EMT logging, you must:

- Include optional code in your monitor at system installation time.
- Supply a program to process the data extracted by the monitor code. This program retrieves extracted data by send/receive calls.

During the system installation dialogue, RSTS/E asks you: EMT Logging? Answer YES to create a monitor that can extract EMT information and pass it to your program.

After creating the proper monitor, you must enable logging for the particular EMT information you want. Use the SET SYSTEM/EMT_LOGGING command to do this. You can use the same command to disable logging for EMT information you no longer want. For example:

```
$ SET SYSTEM/EMT_LOGGING=(NOCRE,NODLN,ASS,DEA)
```

This command disables logging for creating files (CRE) and deleting files (DLN), at the same time enabling logging for assigning devices (ASS) and deassigning devices (DEA). You need the SWCTL privilege to use the /EMT_LOGGING qualifier on the SET SYSTEM command.

The exact logging you choose affects performance. If you choose very common actions, the monitor must take out more time to log them. To keep track of the current EMT logging, use the SHOW SYSTEM/EMT_LOGGING command. This command displays the current logging status of all EMTs. For example:

```
$ SHOW SYSTEM/EMT_LOGGING

Code Mnemonic   Description     Status  Code Mnemonic   Description      Status

CALFIP functions:
   0    CLS    Close a channel  Disabled   2    OPN    Open a channel    Disabled
   4    CRE    Create a file    Disabled   6    DLN    Delete a file     Disabled
   8    REN    Rename a file    Disabled  10    DIR    Directory info    Disabled
  16    RST    Reset channels   Disabled  18    LOK    File lookup       Disabled
  20    ASS    ASSIGN a device  Enabled   22    DEA    DEASSIGN dvice    Enabled
  24    DAL    DEASSIGN all     Disabled  26    CRT    Create .TMP       Disabled
  28    CRB    Create cmpiled   Disabled  30    RUN    Run a program     Disabled

UUO functions:
 -29    TB3    Tables III       Disabled -28    SPL    SPOOL request     Disabled
 -27    DMP    Snapshot dump    Disabled -26    FIL    File utility      Disabled
   .     .          .              .       .     .          .               .
   .     .          .              .       .     .          .               .
   .     .          .              .       .     .          .               .
```

You need no special privilege to use the SHOW SYSTEM/EMT_LOGGING command.

Your logging program is a normal time-sharing job. You can design the program to do such activities as:

- Select pertinent data
- Maintain a log file
- Signal events on one or more terminals
- Control your system while it is running

RSTS/E never logs EMTs for certain jobs, including the Error Logger (ERRCPY) and your EMT logging program itself; in addition the monitor does not allow you to log certain EMTs.

14.5.2 How EMT Logging Works

If EMT logging is available and active, the monitor inspects each EMT as it is received. ("Available" means installed in the system; "active" means that your EMT logging program has properly declared itself as a receiver.) If the newly-received EMT is one that RSTS/E logs, the monitor builds a packet of information about the directive and stores the packet in XBUF. Note that EMT logging does not use small buffers for message transfer.

The packet contains:

- Context data, such as date and time, user job number, and keyboard number
- Directive data, such as information from the job's FIRQB and/or XRB

When the directive is complete, additional information (for example, the RSTS/E error code) is added to the packet. The packet is then made available to your program. Your program can inspect the parameters passed from the user's job to the monitor, determine the relevance of the directive being logged from the packet, and take the appropriate action.

Your program retrieves EMT logging packets by issuing message receive calls. Parameters in the declare receiver call of your program specify:

- How many packets constitute a message
- How many packets can be outstanding at any time
- How much XBUF the EMT logger can use

Each message received by your logging program consists of the packets (one per selected EMT) plus control information. This control information includes a count of EMTs that may have been lost because one or more of the limits you set up at declare time has been exceeded.

See the *RSTS/E Programming Manual* for a description of the use of message send/receive calls as well as specific information about the send/receive calls used for EMT logging.

14.5.3 Data Returned by EMT Logging

The meaning of the data returned by the logging program you write depends on the internal functioning of the monitor. For this reason, both the format and meaning of data returned are subject to change in future releases of RSTS/E. DIGITAL reserves the right to change internal mechanisms and, therefore, makes no commitment to continue providing any specific part of the data described for this release of RSTS/E.

14.5.3.1 Loggable EMTs Enabled or Disabled by SET SYSTEM

For RSTS/E Version 9.7, the following EMTs are available for logging:

`CALFIP functions:`

Code	Mnemonic	Description	Code	Mnemonic	Description
0	CLS	Close a channel	2	OPN	Open a channel
4	CRE	Create a file	6	DLN	Delete a file
8	REN	Rename a file	10	DIR	Directory info
16	RST	Reset channels	18	LOK	File lookup
20	ASS	ASSIGN a device	22	DEA	DEASSIGN dvice
24	DAL	DEASSIGN all	26	CRT	Create .TMP
28	CRB	Create cmpiled	30	RUN	Run a program

```
UUO functions:

        Code  Mnemonic   Description        Code  Mnemonic   Description

         -29    TB3     Tables III           -28    SPL     SPOOL request
         -27    DMP     Snapshot dump        -26    FIL     File utility
         -25    ATR     File attributes      -19    LOG     Set Logins
         -18    RTS     RTS/reslib ctl       -17    NAM     Set RTS name
         -15    ACT     Acctng data          -14    DAT     Date/time ctl
         -13    PRI     Priority, etc.       -12    TB2     Tables II
         -11    BCK     File stats ctl        -9    HNG     Hangup dataset
          -8    FCB     FCB/DDB info          -6    POK     Poke memory
          -3    TB1     Tables I              -2    NLG     Logins <= 1
          -1    YLG     Logins <= max          0    PAS     Create PPN
           1    DLU     Delete PPN             3    MNT     Mount/dismount
           4    LIN     Login                  5    BYE     Logout
           6    ATT     Attach                 7    DET     Detach
           8    CHU     Password/quota        10    ASS     ASSIGN
          11    DEA     DEASSIGN              12    DAL     DEASSIGN all
          13    ZER     Zero a device         14    RAD     Read acctg data
          15    DIR     Directory data        16    TRM     Terminal char
          17    LOK     Wildcard lookup       19    CHE     Cache ctl
          20    CNV     Date => ascii         23    SWP     Swap+ files ctl
          24    JOB     Spawn a job           25    PPN     PPN lookup
          26    SYS     Job status            27    KMC     Connect KMC
          29    STL     Stall system          33    ONX     Open next file
          34    CFG     Configure sys
```

14.5.4 EMT Logging and System Security

The information that EMT logging provides may be helpful in checking and maintaining system security. However, the use of EMT logging as a system security tool depends on many site-specific factors:

- The types of events that can be watched for
- The degree of security required
- The way that EMT logging is set up
- The experience and judgment of the system manager who sets up EMT logging and interprets its data
- The general security practices of the site

In summary, although EMT logging does not guarantee improved system security it can provide a very useful tracking mechanism.

14.6 Monitoring System Status: SYSTAT

During normal timesharing, there are many occasions when you need to monitor the status of your system. RSTS/E provides two system programs for this purpose:

DISPLY
: Automatically displays system status information on your terminal. By default, the program displays the system status every 15 seconds; however, you can specify a different interval. See the following section for a description of the DISPLY program.

SYSTAT
: Displays the status of different parts of the system when you use various program switches. See the *RSTS/E Utilities Reference Manual* for a complete description of SYSTAT and the switches associated with it.

The description of SYSTAT in this section gives a set of guidelines for using the SYSTAT program for more effective system management. Use SYSTAT to:

- Prepare for system shut-down to learn what jobs are active and which disk devices and assignable devices are in use.
- Determine when the number of general small buffers becomes too low. You can then decide whether to use the DCL commands SET SYSTEM/NOLOGINS and SET SYSTEM/LOGINS to restrict the number of users that can log in to the system. Note that you can also use the SHOW BUFFERS command to display the number of available general small buffers.
- Check on the amount of free disk space. The disk status report in SYSTAT reflects the apparent number of free blocks on each disk on the system. For practical purposes, however, such as for allocating a file on the device, all free blocks that SYSTAT reports may not be usable. You may get the message ?No room for user on device when SYSTAT reports there are free blocks available. Note that you can also use the SHOW DISKS command to display the amount of free disk space.

  The file cluster size or the number of clusters you need can prevent a file from fitting on a device. For example, a file whose cluster size is 16 and whose length is 10 blocks may not fit on a device that SYSTAT reports to have 50 free blocks of file space remaining. The cluster size of 16 demands that 16 contiguous blocks of free space exist on the device before the file can be allocated to the device. In some cases, 16 contiguous blocks do not exist on a device and RSTS/E does not allow a file to extend to another physical device even though SYSTAT can give you some indication of the availability of free disk space.
- Follow the progress of user jobs. You can determine if a job is stalled, waiting for resources on your system. If you notice that a run-time value of a job is not increasing (the value is printed in a job status report), it indicates that the job is stalled and waiting for an I/O device. One user job can allocate a device or keep an assignable device locked by having one file open. You can determine who the user is by examining the device status report that associates the busy device with the job number of the user controlling that device. You can then ask the user to free the device or, if that is not possible, you can use DCL commands to force the job off the system (FORCE) or seize the device (ALLOCATE/SEIZE). Note that you can also use the SHOW USERS and SHOW JOBS commands to display user job status.
- Follow the progress of detached jobs. When SYSTAT reports a detached job in the HB (hibernate) state, you know the detached job is attempting to communicate with its terminal. In this case, log in to the system at a free terminal, and then use the ATTACH command to attach the job to the terminal. Once you attach to the detached job, RSTS/E can display the messages from the job. Note that you can also use the SHOW JOBS/DETACHED command to display the status of detached jobs.
- Check on the number of jobs on the system. The free buffer report lists the number of jobs currently logged in to the system and the maximum number of jobs allowed. These numbers are useful when you are adding and removing swap files. After adding the swap files and raising the logins allowed, you can check the free buffer report to confirm the success of the procedure. Before removing a swap file, check this report to make sure that the number of logins allowed is low enough to enable you to remove the swap file at some later time. Note that you can also use the SHOW SYSTEM command to display the number of jobs on the system.

14.7 Dynamic Display of System Status: DISPLY

The DISPLY system program displays the system status on VT52, VT100, and VT200 series terminals. After you start the program and decide how frequently you want it to update the system status on the screen, the program runs until you interrupt its execution. The information DISPLY prints is similar to that of the SYSTAT program. Commands you type during the execution of the program can alter the information displayed on the screen. These commands let you choose only those portions of the system status you need to see.

The program resides in the AUXLIB$: account with the protection code of <232> This protection code lets all users run the program. If you want to restrict running the DISPLY program to users that have the WREAD privilege, change the protection code to <124>.

14.7.1 Running the DISPLY Program

The program prints an identification line and a question after you type the command line RUN AUXLIB$:DISPLY. They appear as follows:

```
$ RUN AUXLIB$:DISPLY
DISPLY V9.7 RSTS V9.7 EDERE
Interval?
```

If you want updates to occur every 15 seconds, press Return to accept the default Otherwise, type the number of seconds you want to elapse between screen updates. You can include any combination of the switches in Table 14–2 in your response. If you select a program switch but do not specify an interval, the program again assumes the default and updates the screen every 15 seconds. Table 14–2 lists the DISPLY program switches.

Table 14–2: DISPLY Program Switches

Switch	Meaning
/[NO]DCA	Indicates if the terminal can use direct cursor addressing to position the cursor on the screen. By default, DISPLY uses /DCA.
/DET	Detaches the job from this terminal or from the terminal you specify.
/KBn:	Prints the output at keyboard unit n if it is available. If you specify the /DET switch, the program runs detached.
/n	Specifies the number of lines to display. The value of n can range from 12 to 24. By default, DISPLY displays 24 lines.
/PRIORITY	Runs the program at a special priority instead of at the normal -8 priority. Requires TUNE privilege.
/[NO]TAB	Indicates if the terminal can correctly process TAB characters. By default, DISPLY uses /TAB.

When you include the /DET switch in response to the INTERVAL question to run the program detached, you interrupt execution and you can use the terminal for other work. As soon as you enter Ctrl/C to interrupt the display, the program prints a message telling you the terminal is available. When you release the terminal by logging out, the program automatically displays the status information on the screen as if it had not been interrupted.

Use Ctrl/C to stop the execution of the DISPLY program. The procedure you follow after stopping the program depends on whether you were running the program attached or detached. If you are running the program attached, entering Ctrl/C to end the program returns control to the keyboard monitor. When running the program detached, entering Ctrl/C causes the program to print the following message before terminating:

```
The terminal is all yours now
```

You must then attach the job running the program to the terminal as follows:

```
HELLO 1,2 <RET>
PASSWORD:
Jobs detached under this account:
  Job  What Size  State   Run-time   RTS
  10   DISPLY 16  SL          5.6    ...RSX
Job number to attach to? <RET>

$ Continue <Yes>? N<RET>
```

When the program attaches to the terminal as a result of the HELLO command, it prints the CONTINUE question. Finally, type NO or any other string not beginning with Y and press Return to terminate the program.

The program displays on your terminal an identification line at the top line of the screen, skips a line, and fills the left portion of the screen with job status information and the right side with information about:

- Busy devices
- Disks
- Run-time systems
- Message receivers
- Free buffers
- Resident libraries

After the screen is full, the program moves the cursor to the first character on the second line of the screen. The program is then idle, waiting to cycle through the display again.

At the specified interval, the program checks the system tables and updates the status information on the screen with any changed data. While executing routines to extract update information, the program prints the message:

```
WORKING...
```

It leaves the cursor to the right of the message. After the update is complete, the cursor returns to its idle position.

While the cursor is in the idle position, you can type commands to modify the contents and arrangement of items on the screen. You can end any command by pressing the Escape key. Although any line terminator works, Escape leaves the cursor positioned on the blank line. The program prints no message or takes no action on invalid statements.

Some commands that add items to the screen take the minus (-) sign as a prefix. A command with a minus sign preceeding it causes DISPLY to delete rather than add a display of the information that would normally appear without the minus sign. That is, the minus sign negates the effect of the command.

Table 14–3 lists the commands you use (usually while the cursor is in its idle position) to delete or add system status information.

Table 14–3: DISPLY Program Commands

Command Type	Format and Description
General	*C* Clears the screen and displays new status.
	Sn Displays memory status in place of job status. Starts with the 8K-word section less than or equal to n. If n is not given, starts at the beginning of memory.
	J Displays job status in the standard manner.
	Jn Displays job status starting with active job n+1. Overcomes physical limitation of the screen.
	Xn Changes the interval to n seconds.
	X0 Updates the display with an interval of 0 seconds (that is, runs continuously) but lowers the priority so that other jobs are not stalled.
Job Status	*T* Displays total CPU time each job has used. The time is displayed as number of hours, minutes, seconds, and tenths of seconds under the RUN-TIME column.
	+ Displays the increment of CPU time each job has used since the display program last updated the screen. The user can return to total CPU time by typing T.
	% Displays the amount of CPU time each job has used as a percent of the total CPU time expended. The user can return to total or increment of CPU time by typing, respectively, T or +.
	J-D Does not display detached jobs.
	J+D Includes detached jobs in display.
	JD Displays only detached jobs.
	J-S Does not display sleeping jobs.
	J+S Includes sleeping jobs in display.
	N Indicates the program name in the WHAT column.
	-N Removes the program name from the WHAT column and replaces it with name of the RTS under which the job is running.

(continued on next page)

Table 14–3 (Cont.): DISPLY Program Commands

Command Type	Format and Description
Job Status (con't)	*P* Indicates, under the Pr column, the exact priority (modulo 8) of the jobs.
	-P Indicates, under the Pr column, the relative priority of the jobs. See the section "Job Status Statistics."
	W Indicates, under the STATE column, the last WAIT state rather than actual state.
	-W Removes the last WAIT state and indicates the actual state of each job.
	K Displays, under the SIZE column, the amount of memory occupied by each job.
	-K Displays, under the SIZE column, the amount of memory remaining for each job.
Disk Structure	*D, Dn, -D* Displays disk structure statistics. If n is 1, places the item first on the screen. A preceding minus sign removes the disk structure statistics from the screen.
	L, -L Displays, under the COMMENTS column, the logical name of each device. The preceding minus sign replaces logical names with standard PUB, PRI, NFS, or LCK notations.
Busy Devices	*B, Bn, -B* Displays busy device statistics. If n is 1, the program places the statistics first on the screen. A preceding minus sign removes busy device statistics from the screen.
Free Buffer	*F, Fn, -F* Displays free buffer statistics. If n is 1, the program places the statistics first on the screen. A preceding minus sign removes free buffer statistics from the screen.
Message	*M, Mn, -M* Displays message receiver statistics. If n is 1, the program places the statistics first on the screen. A preceding minus sign removes message receiver statistics from screen.
Run-Time System	*R, Rn, -R* Displays run-time system data. If n is 1, the program places the data first on the screen. A preceding minus sign removes run-time data from the screen.
Resident Libraries	*H, Hn, -H* Displays resident library data. If n is 1, the program places the data first on the screen. A preceeding minus sign removes the resident library data from the screen.

14.7.2 Screen Layout

The program divides the screen into four major sections:

Header line
Contains the RSTS/E version number, the name of the system, the current date and time of day, and the number of hours, minutes, and seconds since the start of time-sharing operations (termed "up-time"). The header line appears on the first line of the display.

Total statistics line
Tells the percentage of time that is expended by users (User), input and output processing (I/O), and the monitor (Exec) as well as the amount of idle time (Idle) and lost time (Lost). The statistics print on the line below the header line but are replaced at the interval you specify by the WORKING... message. (The statistics are only printed if you use the SET SYSTEM/STATISTICS command.)

Left-hand portion
Contains either the job status statistics or, if you use the S command, the memory usage status.

Right-hand portion
Contains statistics for busy devices, the disk structure, run-time systems, message receivers, system buffers, and resident libraries.

The DISPLY program prints information similar to the information SYSTAT includes in its status reports. The description of the major sections of the dynamic status report includes information about how the DISPLY program reports differ from those of the SYSTAT program. If you need a more complete description of SYSTAT, see the *RSTS/E Utilities Reference Manual*.

The following example displays all four divisions. While the example does not represent an actual log of the DISPLY program, it does give you a picture of how information is displayed. When you run the program on your system, not all the illustrated information will fit on your terminal screen. You will often need to use commands described in Table 14–3 to delete reports so unseen portions of the display become visible. The example lets you see an entire display, unhindered by screen limitations (usually 24 rows).

```
RSTS V9.7-04 NOODLE::        Status on 06-Feb-89 02:10 PM  Up: 118:16:34
    58.6%User,   15.2%I/O,    25.6%Exec,     .2%Idle,    .4%Lost
Job  Who  Where  What Size State Run-time Pr       Busy devices
 1 197,1   Det   ERRCPY  5 SR                Dev Job Why  Dev Job Why
 2   1,2   Det   OPSRUN 23 SL                MM1  26 Open MS0  26 Open
 3   1,2   Det   EVTLOG 18 SL
 4   1,2   Det   PBS... 20 SL                      Disk Structure
 5   1,2   Det   QUMRUN 24 SL              + Dsk Open    Free Clu  Err Comments
 6   1,2   Det   SPLIDL 23 SL                DV0    0      23  1      0 Pri,DLW
 7   1,2   Det   BATIDL 20 SL C01          - DR0    3   14888  8      0 Pri,DLW
 8   1,2   Det   BATIDL 20 SL              - DU0  134  177696 16      0 Pub,DLW
11   1,2   Det   MAILSN 32 SR                DU1    0  194544 16      0 Pri,R-O
12 194,1   KB47  DCL     1 ^C
13   1,2   Det   TWATCH  5 SL              -      Run-time systems
14 202,1   KB46  DCL     4 ^C                ...RSX   0K  16 Mon,KBM
15 248,1   KB44  ...EDT  8 KB          .3    DCL     24K  13 Tmp,KBM
16 248,1   KB45  DCL     4 ^C A08            RT11     4K   1 Tmp,KBM,CSZ
20 206,1   KB49  DCL     4 ^C                BASIC   16K   1 Tmp,KBM,CSZ
22 225,1   KB52  ...EDT  8 KB                BP2      1K   0 Non-res,KBM
24 223,1   KB51  DISPLY 17 RN          .8    FORTH    6K   0 Tmp,KBM,NER
29 208,6   KB59  DCL     4 ^C
30 206,1   KB56J4BACKUP 18 RN         1.2 K-  Gen FIP  Jobs  TTY  Err
31 208,6   KB60J4LINK   28 RN          .3     187 214 30/45    0  3995
35 196,1   KB63  DCL     4 ^C
```

```
      Message receivers
ERRLOG(Prv)      1  0     0/40
OPSER  (Loc)     2  0     0/30
EVTLOG(Prv)      3  0     0/32
EVTLSN(Prv,Nt)   3  1     0/16
QM$CMD(Prv)      4  1     0/20
QM$SRV(Prv)      4  2     0/30
QM$URP(Loc)      4  3     0/10

     Resident Libraries
HISTOG   36K    0 Prm,R/W,NER
CSPLIB    8K    4 Tmp
EDT      39K    1 Tmp
RMSRES    4K    2 Tmp
DAPRES   10K    0 Non-res
BP2RES   20K    0 Tmp
B25SHR    4K    0 Tmp
```

14.7.3 Job Status Statistics

The job status report displays information in eight separate columns on the left-hand portion of your terminal. The report is similar to the information the SYSTAT program prints in its display of job status. Unlike the SYSTAT program, the job status report does not include the name of the run-time system under which the job is running but does include abbreviations indicating the priority of the running job. The titles that the DISPLY program prints at the top of a job status report have the following meanings:

Job — Job number that the system assigns when the job starts timesharing activities.

Who — Account number under which each job runs. If the job is not logged in to the system, **,** appears in this column.

Where — Keyboard number of the job. DET appears in place of the keyboard number for jobs that run detached from the keyboard. The abbreviation KBxJy can appear for a job running on a pseudo keyboard. The value KBx identifies pseudo keyboard unit x; and the value Jy denotes job number y, under which the controlling job is running.

What — Program name that the job is executing.

Size — Current size in K words of the job.

State — Current state of the job indicated by the set of abbreviations in Table 14–4.

Run-time — Hours, minutes, seconds, and tenths of seconds of central processor (CPU) time the job has consumed.

The job status report includes a PR column, which is not displayed by SYSTAT, that identifies the priority of the running job. The PR column can display the following abbreviations:

If -P is in effect:	
+	Higher than normal priority
-	Lower than normal priority
S	Special run priority
^	Ctrl/C temporary priority
K	Keyboard delimiter temporary priority

If P is in effect:	
+n	Positive priority n * 8
0	Zero priority
-n	Negative priority n * 8

Table 14–4 contains a brief description of the abbreviations that can appear in the STATE column of the job status report. The following status descriptions can appear after one or more of the job state abbreviations in Table 14–4:

Lck	Job is locked in memory for the current operation.
Nsw	Job has requested that it not be swapped from memory and cannot be swapped unless it requests additional memory.
Swi	Job is currently being swapped into memory.
Swo	Job is currently being swapped out of memory.
Xnn	Job is swapped out and occupies slot nn in swap file X; file is denoted A, B, C, D to represent files 0 through 3 of the swap structure.

Table 14–4: STATE Column Attributes

Abbreviation	Meaning
??	Job state cannot be determined.
BF	Job is waiting for buffers (no space is available for I/O buffers).
^C	Job is in CTRL/C state, awaiting keyboard monitor input.
CR	Job is waiting for card reader input.
DB,DK,DL,DM, DR,DU,DV	Job is waiting to perform disk I/O.
DX	Job is waiting for floppy diskette I/O.
FP	Job is waiting for file processing action by the system (opening or closing a file, file search).
HB	Job is detached and waiting to perform I/O to or from a terminal.
KB	Job is waiting for input from a terminal.
LP	Job is waiting to perform line printer output.
MT,MM, or MS	Job is waiting for magnetic tape I/O.
RJ	Job is waiting for RJ2780 I/O.
RN	Job is running or waiting to run.
RS	Job is waiting for residency.
SL	Job is sleeping.
SR	Job is sleeping and is a message receiver.
TT	Job is waiting to perform output to a terminal.

14.7.4 Busy Devices

The busy device report lists the devices that are assigned or opened by a specific user. Items reported are the device specification, the job owning that device, and the condition of the device. The disk status information reports assigned disk units. The busy device report prints on the right-hand side of the screen and looks as follows:

```
       Busy  Devices
Dev Job  Why  Dev Job  Why
PK0  11  Open PK1   8  Open
```

The DISPLY program reports the same information as SYSTAT reports in its busy device report, except that DISPLY prints two side-by-side reports to accommodate the limited screen space. Table 14–5 contains the abbreviations that can appear in the WHY column of the busy device report.

Table 14–5: Busy Devices Status Abbreviations: WHY Column

Abbreviation	Meaning
AS	Device is explicitly assigned to a job.
OPEN	Device is open on a channel.
DOS	Magnetic tape is assigned with DOS labeling format.
ANSI	Magnetic tape is assigned with ANSI standard labeling format.

14.7.5 Disks

The disks report describes each disk in use on the system. The report contains the same information that SYSTAT/D displays. The DISPLY program prints the following information; however, because of screen limitations, it does not print column headings or as much detail as SYSTAT does:

- Disk device specification
- Number of open files
- Number of free 512-byte blocks
- Pack cluster size
- Disk hardware error count
- Comments on the status of the disk
- Pack identification name or system logical names (if any) assigned for the devices in use on the system

The disks report appears on the right-hand side of your terminal and looks as follows:

```
      Disk Structure
DR0  0  17252  4   0 Pri,DLW
DR1 42   3044  4   2 Pub,DLW
DR2  0  12496  4   0 Pri,R-O
DR3  5  32144  8   3 Pri,DLW
DR4  1   3272  4   1 Pri,DLW
DR5  0 532032 16   6 Pri,R-O
```

To display the pack identification names or system logical names, you must use the L command described in Table 14–3. This command displays the disk label information in place of the comments that the program normally displays. For example:

```
      Disk Structure
DR0  0  17252  4    0 ROGER
DR1 42   3044  4    2 SYS
DR2  0  12496  4    0 ZEBRA
DR3  5  32144  8    3 ABLE
DR4  1   3272  4    1 MIKE
DR5  0 532032 16    6 YOKE
```

If you want to display the comments again, use the -L command. The limited siz of the screen requires the use of the L and -L commands.

Table 14–6 lists the abbreviations in the COMMENT column of the disks report.

Table 14–6: Disk Status Abbreviations: COMMENT Column

Abbreviation	Meaning
Pub	Cartridge or pack is public.
Pri	Cartridge or pack is private.
NFS	Disk is open as a non-file-structured device.
R-O	Disk unit is read-only (write-locked)
DLW	Date of last write (modify), rather than date of last access, is stored in file accounting entries.
Lck	Disk is in a restricted state.
NFF	New files on this disk are put at the beginning of the directory.

14.7.6 Message Receivers

The message receivers report:

- Provides the job name of the receiving job
- Provides the job number of the receiving job
- Includes the number of messages queued for the job
- Gives the declared maximum number of messages the job can queue
- Tells whether local and network senders are allowed
- Indicates whether local senders must have SEND privilege

The message receivers report appears on the right-hand portion of the screen and often does not appear until you delete the disk and run-time system reports with the -D and -R commands, respectively. A sample of the message receivers report is:

```
Message receivers
ERRLOG(Prv)     1  0    0/40
OPSER (Loc)     2  0    0/30
QUEMAN(Loc)     3  0    0/60
LP0SPL(Prv)     4  0    0/5
BA0SPL(Prv)     5  0    0/5
BA1SPL(Prv)     6  0    0/5
EVTLSN(Prv,Nt)  7  0    0/16
EVTLOG(Prv)     7  1    0/32
```

Due to limited space on the screen, some of the abbreviations are shorter than those printed by the /M switch of SYSTAT. The receiver report does not include the:

- Receiver Identification Block (RIB)
- Object type
- Number of links used and the maximum links allowed.

Table 14–7 describes the abbreviations found in the message receiver status report.

Table 14–7: Message Receivers Abbreviations

Abbreviation	Meaning
Loc	Local senders are allowed for this receiver ID.
Prv	Local senders must have SEND privilege to send to this receiver ID.
Nt	Network senders are allowed for this receiver ID.
1S	Receiver can handle one and only one link.
N1	Both of the above. Network senders are allowed for this receiver ID. Receiver can handle one and only one link.

14.7.7 Free Buffers

The free buffers report provides the following information:

- Number of general small (16-word) buffers not currently in use
- Number of FIP buffers not currently in use
- Number of jobs currently running
- Maximum number of jobs allowed to run
- Number of hung terminal errors
- Total number of errors logged on the system

The report prints the same information the /F switch of the SYSTAT program produces. An example of a free buffers report is:

```
Gen FIP  Jobs  TTY  Err
183   1 14/50    0    9
```

14.7.8 Run-Time Systems

DISPLY shows the run-time systems report on the right-hand portion of the screen. The report includes the same information as SYSTAT/R displays, except for:

- The maximum size in K words that a job running under the run-time system can take
- Some information in the comments column because of limited screen space

The run-time systems report prints:

- Name of each run-time system
- Size of the run-time system in K words

- Number of user jobs currently executing under the control of the run-time system
- Comments regarding the status of the run-time system

A sample run-time systems report is:

```
RSTS V9.7 ERERE::            Status on 31-Jun-89 04:34 PM  Up: 120:08:06
     5.8%User,    2.7%I/O       .5%Exec,   91.0%Idle,     .0%Lost
Job  Who  Where  What Size State Run-time Pr       Busy devices
 1   1,2   Det   ERRCPY  5 SR              +       *** None ***
 2   1,2   Det   OPSRUN 23 SL          .1
 3   1,2   Det   EVTLOG 18 SL                      Disk Structure
 4   1,2   Det   PBS... 21 SL                Dsk Open     Free Clu  Err Comm
 5   1,2   Det   QUMRUN 24 SL              + DV0    0      21  1     0 Pri,
 6   1,2   Det   SPLIDL 23 SL                DR0    3   46640  8     0 Pri,
 7   1,2   Det   BATIDL 20 SL              - DU0  116  175472 16     0 Pub,
 8   1,2   Det   BATIDL 20 SL              - DU9    0  390752 16     0 Pri,
 9   1,2   Det   BATIDL 20 SL              - DU1    0  230320 16     0 Pri,
10   1,2   Det   MAILQ  32 SR              - DU1    0  194544 16     0 Pri,
11   1,2   Det   MAILSN 32 SR
12 223,1   KB43  DCL     4 ^C                     Run-time systems
13   1,2   Det   TWATCH  5 SL              - ...RSX   0K  13 Mon,KBM
14 217,1   KB44  DCL     4 ^C                DCL     24K  10 Tmp,KBM
15 236,1   KB53  DCL     1 ^C                RT11     4K   0 Tmp,KBM,CSZ
16 202,1   KB46  NONAME  2 ^C                TECO    10K   0 Tmp,KBM
18 194,1   KB47  DISPLY 17 RN         1.3    BAS4AL  16K   2 Non-res,KBM
   .
   .
   .
```

Table 14–8 contains a list and description of each abbreviation that DISPLY prints in the COMMENTS column.

Table 14–8: Run-Time Systems and Resident Libraries Report Abbreviations

Abbreviation	Meaning
Non-res	Run-time system or library is nonresident.
Loading	Run-time system or library is being loaded into memory.
Temp	Run-time system or library is removed from memory when not being used.
Perm	Run-time system or library stays in memory when not being used.
Addr:xxx	The value of xxx denotes the starting address of the run-time system or library.
[DF] KBM	Run-time system or library can serve as a keyboard monitor. The optional prefix DF indicates default keyboard monitor.
1US	Run-time system or library can serve only one user.
R/W	Run-time system or library allows read/write access.
NER	Errors occurring within the run-time system or library are not sent to the system error log.
Rem	Run-time system or library is removed from memory as soon as all of its jobs switch to another run-time system or library.
CSZ	Proper job image size (in K words) to run a program can be computed as K-size=(file size+3)/4.
EMT:yyy	Denotes the EMT code for special EMT prefix.

14.7.9 Resident Libraries

The resident libraries report includes:

- Name of the resident library
- Protection code of the resident library
- Size of the resident library
- Number of user jobs currently executing under its control
- Comments regarding the status of the resident library

DISPLY shows the resident libraries report on the right-hand portion of the screen. The report shows the same information as SYSTAT/L displays. Some information in the COMMENTS column may be omitted due to limited space on the screen. See Table 14–8 for a description of the abbreviations used in the COMMENTS column.

14.7.10 Memory Status

The S command causes the program to print a table that shows the use of each 1K-word portion of memory. The memory status report replaces the job status report on the lefthand half of the screen. Use the J command to display the job information again.

Type the S command to have DISPLY print the memory status report as in this example:

```
 Memory usage (Starting at 0K)
  0 MON  MON  MON  MON  MON  MON  MON
  8 MON  MON  MON  MON  MON  MON  MON
 16 MON  MON  MON  MON  MON  MON  MON
 24 MON  MON  MON  MON  MON  MON  MON
 32 MON  MON  MON  MON  MON  MON  MON
 40 MON  MON  MON  MON  MON  MON  MON
 48 MON  MON  MON  MON  MON  MON  MON
 56 *1*  *1*  *1*  *1*  *1*  *1*  *1*
 64 *1*  *1*  *1*  *1*  *1*  *1*  *1*
 72 XBF  XBF  XBF  XBF  XBF  XBF  XBF
 80 XBF  XBF  XBF  XBF  XBF  XBF  XBF
 88 XBF  XBF  XBF  XBF  XBF  XBF  XBF
 96 XBF  XBF  XBF  XBF  XBF  XBF  XBF
104 XBF  XBF  XBF  XBF  XBF  XBF  XBF
112 XBF  XBF  XBF  XBF  XBF  XBF  XBF
120 XBF  XBF  XBF  XBF  XBF  XBF  XBF
128 XBF  XBF  XBF  XBF  XBF  XBF  XBF
136 XBF  XBF  XBF  XBF  XBF  XBF  XBF
144 XBF  XBF  XBF  XBF  XBF  XBF   7
152  7    7    7    7    7    7    7
160  7    7    7    7    7    7    7
```

Due to limited screen space, the program may not be able to display all memory status information simultaneously. The Sn command lets the user determine the starting 8K section. The display program prints the starting section number for the row and eight abbreviations per row. Each of the abbreviations relates to the status of a 1K section of memory. The number of rows the program prints and the extent of memory covered is limited by the terminal. The program indicates the starting 1K section by printing a header line in the following format:

```
MEMORY USAGE (STARTING AT nK)
```

Table 14–9 contains a description of the abbreviations the memory status report uses.

Table 14–9: Memory Status Report Abbreviations

Abbreviation	Meaning
MON	Occupied by the RSTS/E monitor.
n	Occupied by run-time system n, where n is the position in the run-time system list. (A question mark is printed if the run-time system name cannot be determined.)
-n-	Occupied by resident library n, where n is the position in the resident library list.
n	Occupied by job number n.
nLK	Job number n is locked in this 1K portion.
nS	Job number n is being swapped out of memory.
nSI	Job number n is being swapped into memory.
NXM	Memory space is nonexistent.
END	End of physical memory for user jobs.
LCK	Memory is locked.
XBF	Memory is reserved for the extended buffer pool (XBUF).

14.8 Electronic Messaging Systems

RSTS/E can tie in to electronic messaging systems. Electronic messaging systems, such as Telex and TWX, resemble electronic mail systems. However, the messages are not sent between accounts on computers, but between machines (computers or simply terminals). The senders and most of the receivers are subscribers to the electronic messaging service, which sends the messages over the public telephone system.

Every subscribing machine in a messaging system has an identification. When the machine sends or receives messages, it sends this identification to the messaging service to verify its identity. Use the DCL command SET SYSTEM/ANSWERBACK, to store your system's identification. For example:

```
$ SET SYSTEM/ANSWERBACK="221BBAKERSTREET"
```

The identification text must be 24 alphanumeric characters or less. Most messaging services do not allow more than 15 characters. RSTS/E allows spaces and punctuation marks, but most messaging services do not.

You can set any terminal, or any number of terminals, to receive these messages. Use the DCL command SET TERMINAL/ANSWERBACK/PERMANENT to designate a terminal. For example:

```
$ SET TERMINAL KB12:/ANSWERBACK/PERMANENT
```

When messages arrive, RSTS/E stores each of them in a file with a unique name based on the time, for example, 118432.001. The first four digits count the number of minutes since midnight (1184); the next two count the number of seconds since the beginning of the current minute (32). The number in the filetype gives the order in which the message was received. Thus the second message to be received in the same second goes into 118432.002.

RSTS/E keeps all these files in the EMS$: account. You must create an account and assign the logical name EMS$: to it by including this DCL command in your [0,1]START.COM file:

```
$ ASSIGN/SYSTEM dev:[p,pn] EMS$:
```

You can print these files for manual distribution, or an application program can scan them to determine which user on the system should receive which file.

The Unsupported Software package contains a sample procedure under the name EMS.COM.

You can write application programs for other systems that deliver messages to the RSTS/E system in the same manner as an electronic messaging service. To use such a program, you must:

- Have a phone line to a modem connected to the RSTS/E system
- Have the program send Ctrl/E to the RSTS/E system
- Know what identification to expect in response to the Ctrl/E
- Dump an ASCII file containing the message to the RSTS/E system

14.9 DCL Command Descriptions

The following sections describe the DCL commands for the miscellaneous system functions.

14.9.1 LOAD/INDEX Command

The LOAD/INDEX command lets you load SATs into memory during timesharing. By making SATs resident in memory, you can improve the system's performance at the cost of increased memory (XBUF) usage. LOAD/INDEX requires SWCTL privilege.

Format

LOAD/INDEX [device-name]

Command Qualifiers	Defaults
none	none

Prompts

none

Prompts

Command Parameters

device-name

Specifies the disk whose SAT you want to load into memory. If you do not specify a device, the default is the system disk.

14.9.2 LOAD/OVERLAY Command

The LOAD/OVERLAY command lets you load monitor overlays in memory during timesharing. By making certain monitor functions resident in memory, you can improve the system's performance at the cost of increased memory (XBUF) usage. LOAD/OVERLAY requires SWCFG privilege.

Format

LOAD/OVERLAY *overlay-name[,...]*

Command Qualifiers	**Defaults**
/ALL	none

Prompts

Overlay name:

Prompts

Command Parameters

overlay-name

Specifies the monitor overlay that you want to load into memory. The allowable values are:

- ATTRIBUTE—Indicates the monitor overlay that performs file and account attribute read/write operations.
- DCL—Indicates the monitor overlay that performs file-related operations for DCL.
- DELETE_RENAME—Indicates the monitor overlay that performs file deletior and renaming.
- DIRECTORY—Indicates the monitor overlay that performs disk file lookup operations.
- SYSTEM_CALLS—Indicates the monitor overlay that performs all of the following monitor directives:
  - Get monitor tables (parts 1, 2, and 3)
  - Return open file information
  - Date and time conversion
  - Return job status
- TERMINAL—Indicates the monitor overlay code that performs terminal characteristics SET and SHOW operations.

Command Qualifiers

/ALL

Indicates that all possible overlays should be loaded into memory. If you specify this qualifier, RSTS/E does not allow any overlay names.

14.9.3 SET DATE Command

The SET DATE command sets the RSTS/E system date to the value you specify. Optionally, you can specify a new value for the system time. You can specify relative values for the date or time. See the previous section Changing System Date and Time for an explanation of relative dates and times. SET DATE requires the DATES privilege.

Format

SET DATE *date[:time]*

Command Qualifiers	**Defaults**
none	none

Prompts

Date:

Command Parameters

date

Specifies the system date. Insert a space between the date field and the DCL command. You can specify a relative date with the keyword DAYS.

[:time]

Specifies the system time. You can specify a relative time with the keywords; HOURS and MINUTES. Insert a colon (:) between the date and time fields.

NOTE

When you specify the AM/PM suffix, there is no space between the minutes and the suffix. For example:

```
$ SET DATE 5-JAN-90:1:30PM
```

14.9.4 SET SYSTEM Command

The SET SYSTEM command sets the default system characteristics. You need one or more privileges to use this command (see the discussion for each command qualifier).

Format

SET SYSTEM

Command Qualifiers	Defaults
/ANSWERBACK="identification"	none
/DATE_FORMAT=format	See Discussion
/DENSITY=default	See Discussion
/DYNAMIC_REGION_LIMIT=n	none
/EMT_LOGGING=(emt1,emt2,...)	none
/[NO]FMS	none
/HANGUP={DELAYED | IMMEDIATE}	/HANGUP=DELAYED
/HOLD	none
/LABEL=default	See Discussion
/[NO]LAT	none
/[NO]LOG	/LOG
/[NO]LOGINS[=n]	/LOGINS=job max
/MONITOR_NAME=file.name	none
/NAME="installation name"	none
/[NO]PASSWORD_PROMPT[=(type[,...])]	See Discussion
/POWERFAIL_DELAY=n	none
/PSEUDO_KEYBOARDS=n	/PSEUDO_KEYBOARDS=4
/RELEASE	none
/[NO]STATISTICS[=RESET]	/NOSTATISTICS
/SWAP_MAXIMUM=n	none
/TIME_FORMAT=format	See Discussion

Prompts
None

Command Qualifiers

/ANSWERBACK="identification"
Sets the system identification for electronic messaging systems such as Telex or TWX. The exact format of the identification depends on the messaging system you subscribe to. RSTS/E requires that it have no more than 24 characters, all alphanumeric. Most messaging services do not allow identifications longer than 15 characters, or with spaces or punctuation marks. You need SWCTL to use this qualifier.

/DATE_FORMAT=format

Sets the default date format:

- ALPHABETIC—Sets the default date format to alphabetic (DD-MMM-YY)
- NUMERIC—Sets the default date format to numeric (YY.MM.DD)

You need SWCFG privilege to specify this qualifier.

/DENSITY=default

Specifies the magnetic tape default density; n can be any number between 2 and 32766, or the keywords MINIMUM or MAXIMUM may be used. Generally, the available tape drives support 800, 1600, or 6250 bpi densities. If the system default density is less than the drive's lowest possible density, then RSTS/E uses the drive's lowest possible density.

If you do not specify a density for the INITIALIZE, or MOUNT commands, the system chooses the nearest legal density for that tape drive that is not greater than the system default density. If you specify MINIMUM or MAXIMUM for the default density, the system chooses the tape drive's minimum or maximum density for the INITIALIZE, or MOUNT commands.

You need HWCFG privilege to specify this qualifier.

/DYNAMIC_REGION_LIMIT=n

Sets the amount of memory available to use as dynamic regions for those users and applications without the INSTAL privilege. Enter a number n for nK words. The available memory must be shared among all the users and applications on the system, so there may be less than nK available at any given time. You need the SWCFG privilege to use this qualifier.

/EMT_LOGGING=(emt1,emt2,...)

Enables and disables logging for the particular EMT information you want. For example:

```
$ SET SYSTEM/EMT_LOGGING=(NOCRE,NODLN,ASS,DEA)
```

This command disables logging for creating files (CRE) and deleting files (DLN), at the same time enabling logging for assigning devices (ASS) and deassigning devices (DEA). See the section Loggable EMTs Enabled or Disabled by SET SYSTEM earlier in this chapter. You need the SWCTL privilege to use the /EMT_LOGGING qualifier on the SET SYSTEM command.

/[NO]FMS

Enables FMS the next time the system restarts. The /NOFMS qualifier disables FMS. You need the SWCFG privilege to use this qualifier.

/HANGUP={DELAYED | IMMEDIATE}

Sets whether or not the system delays hanging up dial-up lines when carrier is dropped. The default is DELAYED, which means the system waits approximately five seconds before hanging up a dial-up line. Setting IMMEDIATE means the system hangs up a dial-up line as soon as the carrier is dropped. IMMEDIATE is mainly used for European dial-up lines. You need SWCFG privilege to use this qualifier.

/HOLD

Stops execution of all jobs except the one issuing the command. You need HWCTL privilege to use this qualifier.

/LABEL=default

Sets the default magnetic tape label:

- ANSI—Sets the default magnetic tape label to ANSI
- DOS—Sets the default magnetic tape label to DOS

You need SWCFG privilege to use this qualifier.

/[NO]LAT

Enables LAT support the next time the system restarts. The /NOLAT qualifier disables LAT support, saving 2K words of memory. You need the SWCFG privilege to use this qualifier.

/[NO]LOG

Use with the /EMT_LOGGING qualifier. Controls whether or not a confirmation message displays when you enable or disable EMT logging. The default is /LOG (confirmation displays).

/[NO]LOGINS[=n]

Specifies the maximum number of users who can log in to the system at one time. SET SYSTEM/LOGINS=n sets the number of jobs that can be logged in to the system at one time to n. The default for n is JOB MAX. Specify /NOLOGINS if you want to prevent users from logging in to the system by setting the number of logins to one.

/MONITOR_NAME=file.name

Sets the monitor name to the name you specify. The monitor name is a file name and has the usual file name format. The new monitor must reside in _SY0:[0,1] and have a file type of .SIL. The system uses the new monitor at the next automatic restart, or as the default name the next time you start timesharing. You need SYSIO privilege to use this qualifier.

/NAME="installation name"

Sets the system name to the name you specify. You can specify up to 15 characters in the name. You need SWCFG privilege to use this qualifier.

/[NO]PASSWORD_PROMPT[=(type[,...])]

Specifies the types of users that must enter the system password. If you do not specify any types, all users must enter the password. If you want to limit prompting to remote users, use the types NETWORK and DIALUP. You need WACNT privilege to use this qualifier.

/POWERFAIL_DELAY=n

Sets the power fail delay to n. When a power failure occurs, the system waits n seconds before rebooting the system. The value of n can range from 1 to 300. You need SWCFG privilege to use this qualifier.

/PSEUDO_KEYBOARDS=n

Specifies the maximum number, n, of static pseudo keyboards available on the system the next time the system restarts. n can range from 1 to 127. You need the SWCFG privilege to use this qualifier.

/RELEASE

Resumes execution of all jobs on the system. You need HWCTL privilege to use this qualifier.

/[NO]STATISTICS[=RESET]

Turns monitor statistics on and off. The system starts with statistics turned off. Once statistics have been turned on, the SET SYSTEM/NOSTATISTICS command freezes statistics. The SET SYSTEM/STATISTICS command unfreezes them. SET SYSTEM/STATISTICS=RESET resets the statistics counters to zero. You need the SWCTL privilege to use the qualifier.

/SWAP_MAXIMUM=n

Sets the maximum size of the swap file to the value you specify. Enter a number n between 32 and 64 for a size of 32K to 64K words. The system uses the new maximum size at the next restart. You need TUNE privilege to use this qualifier.

/TIME_FORMAT=format

Sets the default time format:

- AM_PM—Sets the default time format to AM/PM time (1:30PM)
- 24_HOUR—Sets the default time format to 24-hour time (13:30)

You need SWCFG privilege to use this qualifier.

14.9.5 SET TIME Command

The SET TIME command sets the RSTS/E 24-hour clock to the value you specify. Optionally, you can set the system date to the value you specify. You can specify relative values for the date or time. See the previous section Changing System Date and Time for an explanation of relative dates and times. SET TIME requires the DATES privilege.

Format

SET TIME *[date:]time*

Command Qualifiers	**Defaults**
none	none

Prompts

Time:

Description

The SET TIME command has no qualifiers.

Command Parameters

[date:]
Specifies the system date. If you specify this optional field, insert a space between the DCL command and the date field and insert a colon between the date and time fields. You can specify a relative date with the keyword DAYS.

time
Specifies the system time. You can specify a relative time with the keywords; HOURS and MINUTES.

NOTE

> When you specify the AM/PM suffix, there is no space between the minutes and the suffix. For example:
>
> ```
> SET TIME 1:30PM
> ```

14.9.6 SHOW SYSTEM Command

The SHOW SYSTEM command displays the system default characteristics as follows:

```
$ show system
System name:                   RSTS V9.7-04 Managers Only
Currently installed Monitor:   BOSSMN          Upon reboot: V97BL3
Swap Maximum:                  64K             Upon reboot: 32K
Date format:                   Alphabetic
Time format:                   AM_PM
Magtape label default:         DOS
Magtape density default:       1600 BPI
Power fail restart delay:      300 seconds
Dynamic Region Limit:          0K words
Hangup:                        Delayed
Monitor Statistics:            Disabled
Job limit:                     13
Current jobs:                  4
Password Prompting:            All users
Last System Password change:   15-Nov-88 at 10:35 AM
LAT is installed
FMS is not currently loaded    FMS will be enabled on reboot
Current Pseudo keyboards:      4               Upon reboot: 6
Answerback message:            None
```

The SHOW SYSTEM command does not require any privileges.

Format

SHOW SYSTEM

Command Qualifiers	Defaults
/EMT_LOGGING	none

Prompts

none

Command Qualifiers

/EMT_LOGGING

Displays the current logging status of all EMTs. You need no special privilege to use this qualifier.

14.9.7 UNLOAD/INDEX Command

The UNLOAD/INDEX command lets you unload SATs that were loaded into memory using the LOAD/INDEX command. UNLOAD/INDEX requires SWCTL privilege.

Format

UNLOAD/INDEX *[device-name]*

Command Qualifiers	**Defaults**
none	none

Prompts

none

Command Parameters

device-name

Specifies the disk whose SAT you want to unload from memory. If you do not specify a device, the default is the system disk.

14.9.8 UNLOAD/OVERLAY Command

The UNLOAD/OVERLAY command lets you unload monitor overlays that were loaded into memory with the LOAD/OVERLAY command. UNLOAD/OVERLAY requires SWCFG privilege.

Format

UNLOAD/OVERLAY *overlay-name[,...]*

Command Qualifiers	**Defaults**
/ALL	none

Prompts

Overlay name:

Command Parameters

overlay-name

Specifies the monitor overlay that you want to unload from memory. The allowable values are:

- ATTRIBUTE—Indicates the monitor overlay that performs file and account attribute read/write operations.
- DCL—Indicates the monitor overlay that performs file-related operations for DCL.
- DELETE_RENAME—Indicates the monitor overlay that performs file deletion and renaming.
- DIRECTORY—Indicates the monitor overlay that performs disk file lookup operations.
- SYSTEM_CALLS—Indicates the monitor overlay that performs all of the following monitor directives:
  - Get monitor tables (parts 1, 2, and 3)
  - Return open file information
  - Date and time conversion
  - Return job status
- TERMINAL—Indicates the monitor overlay code that performs terminal characteristics SET and SHOW operations.

Command Qualifiers

/ALL

Indicates that all possible overlays should be unloaded from memory. If you specify this qualifier, RSTS/E does not allow any overlay names.

Chapter 15

Managing Disks and Tapes

This chapter describes the DCL commands that you use to manage RSTS/E disks and tapes. First, Table 15–1 lists and briefly describes the DCL commands available to you. Then, the chapter provides background information about disk and tape handling on RSTS/E. Finally, the chapter lists each command alphabetically and explains it in more detail.

Some functions of INITIALIZE, MOUNT, and DISMOUNT require privilege. This manual describes all the functions. The *RSTS/E System User's Guide* also describes those functions that do not require any privileges.

Table 15–1: DCL Commands for Disk and Tape Handling

Command	Meaning
INITIALIZE	Initializes a disk or magnetic tape. Use this command to prepare the medium for writing new files.
MOUNT	Logically mounts a disk or magnetic tape. Use this command after you physically mount the disk or tape on the device.
DISMOUNT	Logically dismounts a disk or magnetic tape. Use this command before you physically dismount the disk or tape from the device.

15.1 Working with Disks and Tapes

The system manager or operator is often responsible for mounting and dismounting disk packs and magnetic tapes. Indeed, only a user with MOUNT privilege can load disks on the public structure. Thus, you need to be familiar with commands that prepare tapes and disks for use: INITIALIZE, MOUNT, and DISMOUNT.

15.1.1 Initializing Disks and Tapes

Initializing a disk or tape makes it "like new." You should consider initializing if you have:

- An old disk or tape to recycle; you initialize it to eliminate any existing files that it contains.
- A new disk or tape; you initialize it to perform all the necessary actions that make it usable on your system. (The INITIALIZE command descriptions explain these necessary actions.)

Certain disks require formatting before they are initialized. Formatting writes timing and sense marks onto the disk and destroys any information that the disk contains. You format disks using the DSKINT option of INIT.SYS; for a description see the *RSTS/E Installation and Update Guide.*

You must shut the system down to format the following disks before using them on a RSTS/E system: RK05, RK05F, RP02, RP03, and RX33.

NOTE

> These disks need to be formatted only once. All other disks have been formatted at the factory and need not be formatted again.

The DSKINT option also creates a RSTS/E file structure on the disk. You can later use the DCL INITIALIZE command to clear the disk of files and recreate the RSTS/E file structure.

The INITIALIZE command:

- Has different qualifiers for disk initialization from those for tape initialization.
- Requires WRTNFS privilege when used for disks.

The section "INITIALIZE Command for Tapes" later in this chapter describes tape initialization. See the *RSTS/E System User's Guide* for nonprivileged uses of the INITIALIZE command.

15.1.2 Mounting and Dismounting Tapes and Disks

Whenever you want to use a different tape or disk from that which is mounted, you must logically dismount the current disk or tape (using the DISMOUNT command), and physically remove the disk or tape from the drive. Then you physically mount the new disk or tape on the drive and place the drive on line.

After physically mounting and activating the medium, you must logically mount the new disk or tape (using the MOUNT command). This allocates the drive for appropriate use. For example, a tape drive is allocated for use only through your account. A disk drive with a pack initialized and mounted as public becomes part of the public structure. It can be used by anybody on the system.

The MOUNT command prepares an initialized disk or tape for processing by system commands or user programs. The DISMOUNT command releases a disk or tape that was previously accessed with a MOUNT command. Both MOUNT and DISMOUNT have different qualifiers when used with disks or tapes.

15.2 DCL Command Descriptions

The following sections describe the DCL commands for managing RSTS/E disks and tapes. See Chapter 11 for descriptions of the DCL commands for managing a virtual disk.

15.2.1 DISMOUNT Command for Disks

The DISMOUNT command for disks releases a disk previously accessed with a MOUNT command. You issue this command before you take the drive off line or before you physically dismount the disk. You can dismount any disk with MOUNT privilege. Without MOUNT privilege, you can only dismount a disk mounted /NOSHARE for your job.

The DISMOUNT command deallocates the disk if it was allocated to you. You cannot DISMOUNT a device if there are open files on it. If you try, the system displays the error message:

```
?Account or device in use
```

Format

DISMOUNT *disk-device-name[:] [pack-id]*

Command Qualifiers	Defaults
/PUBLIC	none

Prompts

Device: device-name[:]

Command Parameters

disk-device-name[:]

The physical or logical name of the disk unit containing the pack to be dismounted. If you do not specify a unit number, the following error message appears:

```
?Unit number needed
```

[pack-id]

The pack-id specified when the disk was initialized and mounted. The pack-id is deassigned as a system-wide logical. The pack-id is optional.

Command Qualifiers

/PUBLIC

You need MOUNT privilege to dismount a public disk. (The disk must have been mounted and initialized as public.)

If you do not specify /PUBLIC, the system assumes that the disk was mounted as private. If you try to dismount a public disk without the /PUBLIC qualifier, you get the error message:

```
?Disk is mounted public
```

If you try to dismount a disk by using the /PUBLIC qualifier and the disk was mounted as private, you get the error message:

```
?Disk is mounted private
```

If you do not have MOUNT privilege you can only dismount private disks if they were mounted /NOSHARE. If you try to dismount a public disk, you get the error message:

```
?You must have MOUNT privilege to dismount a public disk
```

If the disk was mounted with the /NOSHARE qualifier, and any unprivileged job except the one to which the disk is assigned attempts to dismount it, the following error message appears:

```
?Account or device in use
```

15.2.2 DISMOUNT Command for Tapes

The DISMOUNT command for tapes releases a tape previously accessed with a MOUNT command. You issue this command before you take the drive off line or before you physically dismount the tape. The DISMOUNT command deallocates the tape drive if it was allocated to you. You cannot DISMOUNT a tape if there are open files on it. Any attempt to do so returns the error message:

```
?Account or device in use
```

When used for tapes, the DISMOUNT command does not require any privileges.

Format

DISMOUNT *tape-device-name[:] [label]*

Command Qualifiers	**Defaults**
/[NO]UNLOAD	/UNLOAD

Prompts

Device: device-name[:]

Command Parameters

tape-device-name[:]

Specifies the physical or logical name of the drive on which the tape is to be dismounted. If you do not specify a unit number with a physical device name, th default is unit 0.

label

Specifies the label on an ANSI tape. Note that it is not necessary to specify a label with the DISMOUNT command. However, if you are using a hard-copy terminal, you can specify a label to keep a log of the tape being dismounted. The label you specify is ignored; it is not checked against the label on the tape.

Command Qualifiers

/[NO]UNLOAD

Specifies whether to unload the tape from the drive. Once a tape is unloaded, it must be manually loaded before it can be mounted again. The default is /UNLOAD.

15.2.3 INITIALIZE Command for Disks

The INITIALIZE command creates a RSTS/E file structure on a disk; that is, it prepares a disk so files can be written to it and read from it. (Be aware that INITIALIZE overwrites any data previously written to the disk unless you specify the /NOEXERCISE/NOERASE qualifiers.)

In creating a RSTS/E file structure for the disk, INITIALIZE writes several structures to the disk. For example, INITIALIZE creates a User File Directory (UFD) for account [0,1] that contains entries for the files SATT.SYS (the storage allocation table) and BADB.SYS (the bad block file).

NOTE

> When you issue the INITIALIZE command, the disk should only be physically mounted in the drive; it should not be logically mounted. You must create DV0: first.

When used for disks, the INITIALIZE command requires WRTNFS, RDNFS, and WREAD privileges.

Format

INITIALIZE *disk-device-name[:] pack-id*

Command Qualifiers	Defaults
/CLUSTER_SIZE=n	n=device cluster size
/DATE=ACCESSED	/DATE=MODIFIED
/DATE=MODIFIED	/DATE=MODIFIED
/[NO]ERASE	/ERASE
/[NO]EXERCISE=n	/EXERCISE=FULL
/INDEX=position	/INDEX=MIDDLE
/MFD_CLUSTER_SIZE=n	n=16
/NEW_FILES=[FIRST | LAST]	/NEW_FILES=LAST
/PRIVATE	/PRIVATE
/PUBLIC	/PRIVATE
/[NO]QUERY	/QUERY
/[NO]RETAIN	/RETAIN
/[NO]WRITE	/WRITE

Prompts

Device:
Pack-id:

Command Parameters

disk-device-name[:]
Specifies the name of the drive on which the disk is physically mounted.

pack-id

One- to six-alphanumeric characters to be used when logically mounting the disk. By default, RSTS/E uses this pack-id as a systemwide logical name for the mounted disk.

Command Qualifiers

/CLUSTER_SIZE=n

Declares the pack cluster size, which is the minimum allocation unit, in 512-byte blocks, for the disk. All files written to this disk use multiples of n blocks, where n is the pack cluster size.

In general, the cluster size specified (n) must be a power of 2 (1, 2, 4, 8, 16), equal to or greater than the device cluster size, but not greater than 64. If you do not specify a cluster size, RSTS/E assumes the device cluster size.

The /CLUSTER_SIZE=n option affects performance. In general, a large pack cluster size speeds disk I/O operations, but wastes disk space. For example, if you declare /CLUSTER_SIZE=16 for a disk, a file consisting of 8193 bytes (16 512-byte blocks plus one byte) requires 2 pack clusters, or 32 blocks. A /CLUSTER_SIZE=8 for the same file would require three clusters, or 24 blocks, of disk storage.

/DATE=ACCESSED
/DATE=MODIFIED

Lets you specify whether to record a file's date of last access (=ACCESSED) or of last modification (=MODIFIED). The default is /DATE=MODIFIED.

/[NO]ERASE

Indicates whether the data on the specified disk is erased. The /NOERASE qualifier applies only if you specify /NOEXERCISE. The default is /ERASE.

NOTE

> Using /NOERASE may cause sensitive data to remain on the disk. If the disk contained files marked as "erase on delete" (protection code of 128 or higher), you should not use /NOERASE unless you are sure that no security problems will result.

Table 15–2 lists the device cluster sizes for the disks that RSTS/E supports.

Table 15–2: Disk Size and Cluster Size

Disk Type	Minimum Device Cluster Size	Acceptable Pack Cluster Size (/CLUSTER_SIZE)	Total Device Size [1](Blocks)
RX50	1	1, 2, 4, 8, 16, 32, 64	800
RX33	1	1, 2, 4, 8, 16, 32, 64	2400
RK05	1	1, 2, 4, 8, 16, 32, 64	4800
RK05F	1	1, 2, 4, 8, 16, 32, 64	4800 [2]
RL01	1	1, 2, 4, 8, 16, 32, 64	10220
RL02	1	1, 2, 4, 8, 16, 32, 64	20460
RD51	1	1, 2, 4, 8, 16, 32, 64	21600
RD52	1	1, 2, 4, 8, 16, 32, 64	60479
RC25	1	1, 2, 4, 8, 16, 32, 64	50902 [2]
RK06	1	1, 2, 4, 8, 16, 32, 64	27104
RK07	1	1, 2, 4, 8, 16, 32, 64	53768
RD31	1	1, 2, 4, 8, 16, 32, 64	41559
RP02	2	2, 4, 8, 16, 32, 64	40000
RP03	2	2, 4, 8, 16, 32, 64	80000
RD32	2	2, 4, 8, 16, 32, 64	83204
RD53	4	4, 8, 16, 32, 64	138668
RM02	4	4, 8, 16, 32, 64	131648
RM03	4	4, 8, 16, 32, 64	131648
RP04	4	4, 8, 16, 32, 64	171796
RP05	4	4, 8, 16, 32, 64	171796
RA80	4	4, 8, 16, 32, 64	237208
RM80	4	4, 8, 16, 32, 64	242575
RD54	8	8, 16, 32, 64	311200
RP06	8	8, 16, 32, 64	340664
RA60	8	8, 16, 32, 64	400175
RM05	8	8, 16, 32, 64	500352
RA81	16	16, 32, 64	888012
RA82	32	32, 64	1216640

[1] Size when disk is initialized at lowest acceptable cluster size

[2] For each unit; 2 units per drive

/EXERCISE=n
/NOEXERCISE

Controls whether to check for bad blocks by "exercising" the disk. This means that the number of patterns you select (n) are written to the disk, and then each block is read to make sure that the patterns were written correctly. If the patterns read do not match the patterns written, RSTS/E assumes the block is bad. RSTS/E adds any bad blocks found to the "bad block" file (BADB.SYS in account [0,1] on the disk). RSTS/E does not store data in any blocks allocated to the BADB.SYS file.

You can have the INITIALIZE command run from zero to three patterns by specifying 0, 1, 2, or 3 for n. Each pattern consists of three octal words; the pattern used for each of these numbers is:

Pattern	Three Octal Words
1	155555 133333 066666
2	133333 066666 155555
3	066666 155555 133333

For example, /EXERCISE=2 causes pattern 1 to be written to and read from the disk, followed by pattern 2 being written to and read from the disk.

You can also select the default, /EXERCISE=FULL, which uses all three patterns, one at a time. That is, each block is written to and read from the disk three times, each time with a different pattern. Using the FULL argument increases the probability that all bad blocks are found, and decreases the probability that you will lose information later by writing into a bad block. (Note that /EXERCISE=FULL is equivalent to /EXERCISE=3.)

The /NOEXERCISE qualifier indicates that you do not want to check for bad blocks. For example, if you previously initialized the disk and feel that the current bad block file is accurate, you can save time by specifying the /NOEXERCISE qualifier. This still writes one pattern all over the disk unless you specify /NOERASE, but it does not read it back in. (Note that /NOEXERCISE is equivalent to /EXERCISE=0.

/INDEX=position

Positions the SATT.SYS file on the disk. SATT.SYS is the storage allocation table created in account [0,1] during initialization.

The position argument can be:

BEGINNING	Puts SATT.SYS at the beginning of the disk to avoid fragmenting the disk. That way, more contiguous space is available to store other files on the disk. This is the default position for SATT.SYS for disks of 4500 blocks or less if you do not specify the /INDEX option.
MIDDLE	Puts SATT.SYS in the middle of the disk. For moving-head disks, it helps to locate SATT.SYS near the middle of the disk, which reduces average seek times for the disk heads. MIDDLE is the default for disks of more than 4500 blocks if you do not specify the /INDEX option.
n	Puts SATT.SYS at device cluster number n, where n can range from 1 to the total device size divided by the device cluster size. See Table 15–2 for these values.

/MFD_CLUSTER_SIZE=n

Declares the minimum allocation unit, in 512-byte blocks, for the Master File Directory (MFD). The MFD is a special "catalog" structure that RSTS/E creates during disk initialization. The INITIALIZE command puts the MFD at the beginning for disks of 4500 blocks or less, or at the middle for disks of more than 4500 blocks.

Along with other key structures, RSTS/E updates the MFD each time users add or delete accounts and files on the disk. Because RSTS/E must access the MFD frequently, you increase performance when the MFD cluster size is large. On the other hand, a small MFD cluster size can save some disk space.

The MFD cluster size (n) can be 4, 8, or 16; it must be greater than or equal to the value entered for the CLUSTER_SIZE qualifier, unless the CLUSTER_SIZE is 32 or 64; in that case, the MFD cluster size remains 16. The default is /MFD_CLUSTER_SIZE=16.

/NEW_FILES

Indicates whether files stored on the disk should be placed at beginning of the account (/NEW_FILES=FIRST) or at the end of the account (/NEW_FILES=LAST). The default is /NEW_FILES=LAST.

/PRIVATE

Allows only users who have accounts on a disk to access it. If you do not specify /PUBLIC or /PRIVATE, then RSTS/E assumes /PRIVATE.

You can create a system disk by initializing the disk as /PRIVATE and then using the CREATE/ACCOUNT command to create and position accounts [1,1] and [1,2]. In this case, note that you must transfer files to the disk and use the HOOK utility to make it a RSTS/E system disk.

/PUBLIC

Allows anyone with an account on the system to access the disk.

You can use /PUBLIC to identify any disk on the system as public, except the system disk. When the disk is mounted, it is considered part of the public structure. You need MOUNT privilege to mount a public disk. After you mount it, the disk is a logical extension of the system disk; thus, any user can create files on it.

In general, DIGITAL recommends against using /PUBLIC, because it can degrade performance over time. Consider the following points if you decide to select /PUBLIC for disks on your system:

- If your system disk is nearly full, and you want to dedicate one or more additional disks to general file storage, you might add public disks to your system. Users are not generally aware of whether their files are on the system disk or a public disk.
- A public disk is an extension of the system disk; therefore, it should always remain mounted during timesharing. In addition, you must remount it every time you restart the system. If you dismount the disk during timesharing, some users' files will mysteriously (to them) disappear. Dismounting a public disk causes disruption unless you transfer all of the files on the disk to the system disk or to some other public disk.
- Before adding a public disk to your system, consider the alternative of adding a private disk for use by specific users. DIGITAL recommends this alternative because it yields significantly better system performance. However, this means that you must create accounts explicitly on the private disk. In addition, users who have accounts on the disk must refer explicitly to that disk in file specifications.

Because of these considerations, the system provides several safety checks on initializing and mounting a public disk. First, you must have MOUNT privilege to do so. Second, you must specify /PUBLIC two times: when you initialize the disk, and when you mount the disk. Third, you must specify when you want a file placed on DV0:, the virtual disk; the system never puts files on DV0: automatically.

/QUERY
/NOQUERY

When you use the /QUERY qualifier, INITIALIZE displays the characteristics of a disk you have selected that already has a RSTS/E file structure, and asks if yo still want to initialize the disk.

Thus, /QUERY lets you verify that you are initializing the disk you intended, thereby keeping you from accidentally wiping out the wrong disk. If you specify /NOQUERY, INITIALIZE simply displays the characteristics of the disk before overwriting its contents. The default is /QUERY.

The display of disk characteristics is similar to:

```
This disk pack appears to be a RSTS/E formatted
disk with the following characteristics:

Pack ID :             THELMA
Pack Cluster Size :   4
Pack is currently :   Private,
                      Level 1.2

Proceed (Y or N)?
```

You must respond to the prompt with YES or NO, or some abbreviation of these. NO cancels the initialization; YES requests that it proceeds. There is no default response to the prompt.

/RETAIN
/NORETAIN

Controls whether to use a previously initialized disk's bad block file. A disk that has not been previously initialized has no bad block file. In this case, /RETAIN and /NORETAIN are meaningless, and are ignored if specified. /RETAIN is the default on a previously initialized disk.

If you use both the /RETAIN and /EXERCISE qualifiers, the INITIALIZE command keeps the old bad block file and adds any additional bad blocks it find while it is checking the disk for bad blocks.

If you specify /NOEXERCISE, then you should use /RETAIN. Because /NOEXERCISE does not exercise the disk for bad blocks, you should use /RETAI to keep the existing bad block information.

The /NORETAIN qualifier is useful if problems with a disk drive have been causing apparent bad blocks on a disk. After you learn that the problem is with the drive, you can recover space on the disk by using /NORETAIN.

/WRITE
/NOWRITE

Determines whether the disk should default to read-only (/NOWRITE) or read/write (/WRITE) access when it is mounted. This is useful when you initiali a disk that is usually mounted read-only. The default is /WRITE.

When mounting a disk, you can override the read-only access by specifying the /WRITE qualifier with the MOUNT command.

15.2.4 INITIALIZE Command for Tapes

The INITIALIZE command for tapes prepares a new tape or recycles a used tape that contains no useful files. The INITIALIZE command for tapes makes any existing data on the tape unavailable to file-structured access. For ANSI tapes, it also writes a label that RSTS/E uses to identify the tape in a MOUNT command (ANSI tapes require a label). INITIALIZE allocates the tape drive if it is not already allocated.

When used for tapes, the INITIALIZE command does not require any special privilege unless the device is restricted. If so, DEVICE privilege is required.

Format

INITIALIZE *tape-device-name[:] [label]*

Command Qualifiers	**Defaults**
/DENSITY=nnn	See Discussion
/FORMAT=ANSI	See Discussion
/FORMAT=DOS	See Discussion

Prompts

Device: magtape[:]
Label: label (if ANSI)
Proceed (Y or N)?

Command Parameters

tape-device-name[:]
Specifies the name of the drive on which the tape is physically mounted.

[label]
Specifies the identification label to be encoded on the tape. The label can consist of a maximum of six alphanumeric characters.

An ANSI-format tape requires a label; RSTS/E prompts you for a label if you do not specify one. RSTS/E checks this label against the label you specify when you later use the MOUNT command (see the section "MOUNT Command for Tapes").

DOS tapes do not allow labels: RSTS/E ignores any label you specify for them and displays a warning message.

Command Qualifiers

/DENSITY=nnn
Specifies the density in bits per inch (bpi). The argument nnn can be any number between 2 and 32766, or the keywords MINIMUM or MAXIMUM may be used. Generally, available tape drives support 800, 1600, or 6250 bpi.

If you specify MINIMUM or MAXIMUM, the system checks the tape drive and sets the density to the minimum or maximum the drive supports. If you specify a density that the tape drive does not support, RSTS/E displays an error message and aborts the command. If you do not specify a density, the default is the system default density.

/FORMAT=ANSI
/FORMAT=DOS

Specifies the tape format. If you do not specify a format, the system uses the current system default (SHOW SYSTEM command displays the current system defaults). You can change the default format using the DCL command, SET SYSTEM/LABEL= (see Chapter 14). As mentioned earlier, ANSI format requires a label; DOS format ignores any label specified.

15.2.5 MOUNT Command for Disks

The MOUNT command prepares an initialized disk for processing under timesharing. To mount a disk, you must:

1. Physically mount the disk on a device drive
2. Put the drive on line
3. Use the MOUNT command to logically mount the disk

You need the MOUNT privilege to mount a disk that has been initialized as public. For users with MOUNT privilege, the MOUNT command rebuilds a "dirty" disk – one that has been physically dismounted (removed from the drive) or taken off line without being logically dismounted first with the DISMOUNT command. This is the only time that a disk's "dirty" bit is set, although you may choose to rebuild the disk at other times to ensure integrity of its structure.

If MOUNT discovers the disk needs rebuilding, it displays the message:

```
Disk is being rebuilt - wait...
```

The rebuild operation proceeds and the disk is logically mounted when the dollar prompt ($) for the next DCL command appears. However, if the rebuild operation discovers blocks that have been allocated to more than one file (doubly allocated blocks), MOUNT displays a message asking which file to allocate each doubly allocated block to. You can choose a file at that point or stop and mount the disk without rebuilding, which lets you examine the situation further.

Format

MOUNT *disk-device-name[:] pack-id [logical-name[:]]*

Command Qualifiers	Defaults
/OVERRIDE[=identification]	/none
/PRIVATE	/PRIVATE
/PUBLIC	/PRIVATE
/[NO]QUOTA	See Discussion
/[NO]REBUILD	See Discussion
/[NO]RESTRICT	/NORESTRICT
/[NO]SHARE[=n]	See Discussion
/[NO]WRITE	See Discussion

Prompts

Device: device-name[:]
Pack-id: pack-id

Command Parameters

disk-device-name[:]

Specifies the physical or logical name of the drive containing the disk you want to logically mount.

pack-id

Specifies the one to six character alphanumeric pack identification label written to the disk during initialization. (See the INITIALIZE command description in the section "INITIALIZE Command for Disks.") MOUNT then verifies that the pack-id written on the disk is the same as the label you specified in the MOUNT This process helps make certain that you physically mounted the right disk.

You must include a pack-id to mount a disk, unless you have the MOUNT privilege and use the /OVERRIDE qualifier. If you use this qualifier, do not ente the pack-id parameter, but only the device-name parameter and the logical-nam parameter (if any).

[logical-name[:]]

Lets a user with INSTAL privilege assign a system-wide logical name to the dis drive specified by a device name. If you do not specify a logical name and you have INSTAL privilege, the system uses the pack-id label as a system-wide logic name for the drive. You may want to use this logical name to keep the pack-id from unauthorized users. Unauthorized users can see logical names assigned to devices in SYSTAT. RSTS/E ignores the specification of a logical name by an unauthorized user (without INSTAL privilege), but the mount succeeds.

Command Qualifiers

/OVERRIDE[=identification]

Eliminates the need to specify the pack-id when mounting the disk. The identification argument is optional and does not change the meaning of the qualifier. Yo must have MOUNT privilege to include this qualifier. If you include /OVERRID then MOUNT assumes that the second parameter, if specified, is the logical nam for the device instead of the pack ID.

By default, you must include the pack ID of the disk.

/PRIVATE
/PUBLIC

Declares that the disk you are mounting is accessible to all users on the system (public disk), or only accessible to users who have accounts on the disk (a private disk). The default is /PRIVATE.

The action taken for /PRIVATE and /PUBLIC depends on two things:

- Whether you have MOUNT privilege when issuing the MOUNT command
- Whether you initialized the disk as public or private

In general, the disk is safeguarded, even against a user with MOUNT privilege, from being inadvertently made available for public use.

For Users with MOUNT privilege

If you have MOUNT privilege, you can MOUNT a disk initialized as private. Yo can explicitly declare the usage with the /PRIVATE or /SHARE qualifier, but thi is not required. You can also mount the disk with the /NOSHARE=n qualifier, t make the disk accessible only to job n (see the discussion for /NOSHARE[=n]).

Table 15–3 summarizes the possibilities of the /PUBLIC, /PRIVATE, and /[NO]SHARE qualifiers. (If none of these qualifiers is specified, RSTS/E assumes /PRIVATE.)

Table 15–3: The /PUBLIC, /PRIVATE, and /[NO]SHARE Qualifiers for the MOUNT Command

	INITIALIZE for disk was:	
Privileges	**Public**	**Private**
User does not have MOUNT privilege	Any MOUNT returns an error.	Only a MOUNT/NOSHARE succeeds.
User has MOUNT privilege	MOUNT with /PUBLIC succeeds: the disk is accessible to all users. MOUNT with /PRIVATE or /SHARE succeeds: the disk is accessible to only those users with an account on the disk. MOUNT with /NOSHARE[=n] succeeds; the disk is accessible only to job n (the user's job, by default). MOUNT with no qualifier succeeds as private disk: the disk is accessible to only those users with an account on the disk: RSTS/E displays a warning message.	MOUNT/PUBLIC returns an error. MOUNT with no qualifier or with /PRIVATE, /SHARE, or /NOSHARE succeeds.

/QUOTA
/NOQUOTA

Controls whether quota checking is performed when the disk is mounted. This qualifier applies only to Level 1.2 disks. The default is /QUOTA. DV0: is always /NOQUOTA.

/REBUILD
/NOREBUILD

Controls the rebuilding of a disk, regardless of whether the disk was initialized as read/write or read-only. A dirty disk is one that has been physically dismounted (removed from the drive) without being logically dismounted first by a DISMOUNT command.

A DISMOUNT checks to see that no files are currently open on a disk. If files are open, the DISMOUNT does not succeed. If someone makes a mistake and physically removes a disk pack from a disk drive while programs are still using files on the disk, the usual "clean-up" operations that the RSTS/E monitor performs may not have occurred. For example, the monitor keeps the file SATT.SYS (the storage allocation table for the disk) in memory and updates it when files are added to or deleted from the disk. If someone physically dismounts a disk before the monitor has written SATT.SYS back to disk, the data in the old SATT.SYS will probably be incorrect.

Rebuilding a disk requires MOUNT privilege. This operation performs necessary clean-up before the disk can be used again. Specifically, a rebuild operation:

- Locates blocks that have been allocated to more than one file. The system displays a message listing the files to which such blocks have been allocated, and asks you to delete all but one of the files.
- Deletes invalid directories.
- Deletes all files that have the type .TMP, are marked for deletion, or have no accounting entry.

- Builds a new storage allocation table (SATT.SYS), to show current file allocations (after deletions accomplished by rebuilding).
- Zeroes all blocks that were in the old storage allocation table (SATT.SYS), bu not in the new one. This is done as a security precaution: in case these block belonged to files with a protection code of <128> or higher (which are always zeroed when they are deleted).

A MOUNT operation for disk automatically checks to see if the disk is dirty. If so, and you have MOUNT privilege, a rebuild occurs automatically, regardless o whether you specify /REBUILD. The following message appears to indicate that the disk is being rebuilt:

```
Disk is being rebuilt - wait ...
```

However, if a user without MOUNT privilege tries to mount a dirty disk, RSTS/ displays the error message:

```
?Disk needs rebuilding but you do not have MOUNT privilege
```

The /REBUILD qualifier forces a rebuild on a disk. It lets the user with MOUN privilege rebuild a disk that the system has not identified as "dirty." For exampl some programs leave temporary files in the user's account, rather than deleting them before the program exits. Such a disk is not flagged as dirty. If you want to get rid of the temporary files on a disk, you can mount the disk using the /REBUILD qualifier.

If you specify /REBUILD and the drive is write-protected, RSTS/E displays the following error message:

```
?Can't rebuild disk because device is write-protected
```

If a user without MOUNT privilege specifies /REBUILD, RSTS/E displays the following error message:

```
?You must have MOUNT privilege to rebuild the disk
```

The /NOREBUILD qualifier overrides an automatic rebuild of a dirty disk. You may not want to take the time to rebuild the dirty disk. Or, the disk may have doubly allocated blocks, and you do not want to risk deleting them. In this case, you can mount the disk with the /NOREBUILD qualifier (thus granting only read-only access) and try to correct the situation. For example, you could use the COPY command to copy the files to new locations (where they do not contai doubly-allocated blocks) and try to sort out the situation from there.

If you have MOUNT privilege, and specify /NOREBUILD to mount a dirty disk, the disk is mounted restricted and read-only, and two warning messages appear

```
%Disk needs rebuilding
%Disk is restricted and mounted no-share, read-only
```

Thus, the disk is accessible only to the job that mounted it.

When a user without MOUNT privilege specifies /NOREBUILD to mount a "clean" (rebuilt) disk, the mount succeeds and the disk is uneffected.

/[NO]RESTRICT

Indicates whether the disk is restricted. If you specify the /RESTRICT qualifier in the MOUNT command line, only users with DEVICE privilege can access the specified disk. The default is /NORESTRICT.

/SHARE
/NOSHARE[=n]

Controls whether to limit disk access to:

- Job n (/NOSHARE=n)
- The job that mounted the disk (/NOSHARE)
- Any user with an account on the disk (/SHARE)

A disk mounted as shared is the same as a disk mounted as private. A user with MOUNT privilege can mount a private or public disk as shared or nonshared.

If you do not specify /PRIVATE, /PUBLIC, or /[NO]SHARE, then the default action is to mount the disk as private (shared).

The /NOSHARE qualifier conflicts with either /PUBLIC or /PRIVATE. You can use /SHARE with either /PUBLIC or /PRIVATE; its presence in the same command line as the other two qualifiers has no effect.

/WRITE
/NOWRITE

Controls whether data can be written to the mounted disk.

If you specify /NOWRITE, the disk is write-protected, which means that no users can write to the disk. The /NOWRITE qualifier protects files on the disk; they can only be read from, not written to.

If you specify /WRITE, users who access the disk can write to it.

When you do not specify /WRITE or /NOWRITE, the default depends on whether the disk was initialized as read/write or read-only, and whether the disk is clean or dirty. For disks initialized as read-only, the default is to give read-only access (/NOWRITE) to the disk. For disks initialized as read/write, there are a few more conditions to consider. If the disk is clean or can be rebuilt (/NOREBUILD was not specified), and the drive is not write-locked, /WRITE is the default. If the disk is clean and the drive is write-locked, /NOWRITE is the default, and the following warning message appears:

```
%Device write protected
```

If the disk is dirty and /NOREBUILD was specified, the disk is mounted read-only, noshare.

15.2.6 MOUNT Command for Tapes

The MOUNT command for tapes verifies that the tape is properly loaded onto th unit and checks an ANSI tape for the label specified in the MOUNT command (ANSI tapes must have labels). You issue a MOUNT command after you have physically mounted a tape on a tape unit and placed it on line.

If the device is restricted, you need DEVICE privilege to use the MOUNT command.

Format

MOUNT *tape-device-name[:][label]*

Command Qualifiers	Defaults
/DENSITY=nnn	See Discussion
/FORMAT=argument	See Discussion
/OVERRIDE[=identification]	none
/[NO]WRITE	See Discussion

Prompts

Device: device-name[:]
Label: label

Prompts

Command Parameters

tape-device-name[:]
Specifies the physical or logical name of the drive on which the tape is physicall mounted. If you do not specify a unit number with a physical device name, RSTS/E assumes unit 0.

[label]
This parameter is necessary only for ANSI tapes. If you try to mount an ANSI tape without specifying a label, RSTS/E prompts you for a label.

A label is not required (and is ignored) when you mount a tape in DOS or foreig format. In this case, RSTS/E displays a warning message:

```
%Label ignored
```

The label specifies the one- to six-character alphanumeric label written to the tape when it was initialized (see the INITIALIZE command description in the section "INITIALIZE Command for Tapes"). MOUNT then verifies that the labe on the tape is the same as the one you specified in the MOUNT command. This process helps make certain that you have mounted the right tape.

Command Qualifiers

/DENSITY=nnn
Specifies the density in bits per inch (bpi). The argument nnn can be any numb between 2 and 32766, or the keywords MINIMUM or MAXIMUM may be used. Generally, available tape drives support 800, 1600, or 6250 bpi.

If you specify MINIMUM or MAXIMUM, the system checks the tape drive and sets the density to the minimum or maximum the drive supports. If you specify a density that the tape drive does not support, RSTS/E displays an error message and aborts the command. If you do not specify a density, the default is the system default density.

/FORMAT=ANSI
/FORMAT=DOS
/FORMAT=FOREIGN

Indicates whether the tape is in a standard format used by the RSTS/E operating system. The default format is the current system default (see SHOW SYSTEM, Chapter 14).

The *RSTS/E System User's Guide* describes ANSI and DOS formats. A tape is FOREIGN if it is not in ANSI or DOS format.

NOTE

> If you mount a tape with /FORMAT=FOREIGN, the program you use to read the tape must know what types of labels there are on the tape or be able to process any labels on the tape.

/OVERRIDE[=identification]

Lets you mount ANSI magnetic tapes without specifying the label name. Since DOS format tapes have no label, the /OVERRIDE qualifier is ignored if the tape format is DOS. The identification argument is optional and does not change the meaning of the qualifier. You need no privileges to use the /OVERRIDE qualifier.

/WRITE
/NOWRITE

RSTS/E checks to see whether the tape is read/write (the write ring is present) or read-only (the write ring is not present).

If the write ring is not present and you specify /WRITE, RSTS/E displays the following error message:

```
?Device is write protected
```

If the write ring is not present and you do not specify either qualifier, RSTS/E displays the following warning:

```
%Device is write protected
```

Chapter 16

Backing Up and Restoring Files

Backing up disk files and accounts means copying the contents of the disk to another volume or set of volumes. Backing up disks is a precautionary measure to let you recover from the loss or destruction of valuable information.

Most sites establish a policy and a schedule for regularly backing up disk files. This chapter provides the operating procedures for backing up both selected files and entire disks.

It is just as desirable to back up information on private disks as it is to back up public disks. However, responsibility for backing up the files on private disks could be left to the individual owners of those files and disks.

There are two kinds of backups of disk files:

- Incremental, or partial backups
- Full, or all-inclusive, backups

In either case, the backup medium can be disk or magnetic tape. Incremental backups save only those files that have been modified recently. Periodic full backups are necessary to provide the basis for reconstruction of an entire disk.

As a rule, incremental backups are undertaken more frequently than full backups. After consulting with users of the system, you decide how often to back up files and volumes and how long to retain backup media.

Generally, you are responsible for setting up a schedule for backing up files and accounts, and for maintaining this schedule. The following schedule for backing up public disk volumes on magnetic tape affords adequate protection of data for many installations:

- Daily—An incremental backup retained for seven days. This schedule requires seven daily tapes (or sets of tapes) that are rotated once a week.
- Weekly—An incremental backup retained for four weeks. This schedule requires four weekly sets of tapes that are rotated once every four weeks.
- Monthly—An all-inclusive backup retained for a year. This schedule requires twelve monthly sets of tapes that are rotated once a year.

Despite all precautions, there is always the risk of losing a file. Frequent backup and longer retention periods reduce this risk. You may also want to consider an off-site storage facility for the monthly save sets.

You can perform full backups to magnetic tape or to another disk. Each has its advantages and disadvantages. The advantage of using magnetic tape for backups is the much lower media and storage cost, which may permit you to retain backups longer than keeping full backups on disk.

However, there are several advantages to keeping copies on disk, which in some cases outweigh the higher cost. Disks exhibit better data reliability than magnetic tapes. Furthermore, disks tend to degrade less in storage.

16.1 Recovering Your System Disk (RECOVR)

You can use the BACKUP utility and the command file RECOVR.COM to recreate your system disk in case it becomes corrupted. RECOVR.COM is located in account [0,1].

BACKUP lets you backup your system disk once it is ready for normal timesharing; RECOVR.COM creates a bootable disk or tape with the minimum files needed to start your system. Use them in the following procedure to create a recovery medium and restore your system disk:

Creating the recovery medium:

1. After the system is installed and running, use the DCL BACKUP command to create a full backup of all accounts and files on the system.
2. Once the backup is finished, use the command file RECOVR.COM to create the bootable disk or tape from which the RESTORE command can recreate the system disk.
3. Perform system backups on a regular basis.

Recreating the system disk:

1. Boot the recovery disk or tape you created with the RECOVR.COM file. See Chapter 2 for a description of this procedure.
2. Begin installation procedures. Note that this will reinitialize the system disk.
3. After you answer "yes" to the "Start timesharing?" prompt, the system displays a message stating it is ready for you to begin using the RESTORE command.
4. At the DCL prompt, physically mount your latest full backup set. Enter the RESTORE/ACCOUNTING command.
5. Next, mount the latest incremental backup (if you do incremental backups) and enter the RESTORE/ACCOUNTING/REPLACE command.
6. When the restore completes, run the SHUTUP program and restart the system using your normal monitor (restored from the backup set).

The system disk is now back to the point of the last backup.

16.2 DCL Command Descriptions

You can use two DCL commands: BACKUP and RESTORE. Files created by BACKUP are called Backup sets. A Backup set is a RSTS/E file that can be copied just like any other RSTS/E file. However, only BACKUP and RESTORE can interpret the data in a Backup set. The rest of this chapter describes BACKUP and RESTORE in detail.

NOTE

> Because the BACKUP and RESTORE operations can use significant amounts of system resources, you need the TUNE privilege to use these commands. Additional privileges may be required to use certain qualifiers (for example, /ACCOUNT_DATA requires GACNT or WACNT privilege). If you want to delegate the responsibility for backing up certain files (for example, files on private disks) to the individual owners, they will need GREAD or WREAD privileges.

16.2.1 BACKUP Command

The BACKUP command creates a Backup set on a specified disk or magnetic tap from one or more RSTS/E disk files. Once the BACKUP utility determines the syntactical validity of the command specification and issues any warning messag relative to either the command or the environment, BACKUP prompts with:

```
Please mount volume 1 of Backup set backup-set-name
Where can this volume be located <default>?
```

At the prompt, enter the device name to write the Backup set to. The default is the output device you specified in the BACKUP command line. You can use the /INITIALIZE and /NOINITIALIZE qualifiers to control initialization of the outpu device.

Format

BACKUP *input-file-spec output-backup-set-spec*

Command Qualifiers	**Defaults**
/[NO]ACCOUNT_DATA	/ACCOUNT_DATA
/BLOCK_SIZE=value	/BLOCK_SIZE=2048
/BRIEF	none
/BUFFER_SIZE=n	See Discussion
/CREATED=(date/time or range)	ALL if neither /CREATED
/MODIFIED=(date or range)	or /MODIFIED is present
/[NO]CRC	/NOCRC
/DENSITY=n	Highest available
/DIRECTORY	none
/END=[NO]REWIND [,[NO]DISMOUNT]	/END=DISMOUNT
/EXCLUDE[=(filespec list)]	None excluded
/[NO]FILE_DATA	/FILE_DATA
/FULL	none
/[NO]GROUP_SIZE[=n]	/GROUP_SIZE=10
/INCLUDE=(filespec list)	None included

/IN_LABEL=name	Volume ID is not checked
/[NO]INITIALIZE	See Discussion
/[NO]LIST_FILE[=filespec]	/LIST_FILE=KB:
/OUT_LABEL=name	See Discussion
/[NO]OUTPUT[=filespec]	/OUTPUT=KB:
/[NO]PROMPT	/PROMPT
/[NO]QUERY	/NOQUERY
/[NO]REWIND	/NOREWIND
/SELECT=(filespec list)	See Discussion
/[NO]VERIFY	/NOVERIFY

Prompts

See Discussion

Restrictions

Backup sets produced by the BACKUP utility are not bootable. Reboot using a recovery medium (for example, a disk or tape produced by RECOVR.COM or the latest installation kit) then use the RESTORE command to restore the backup set.

Command Parameters

input-file-spec

Identifies the files to be backed up. The file specification should be a standard RSTS/E file specification or a list of specifications separated by the plus (+) sign or a comma (,):

dev:[p,pn]filename.typ,...

You must specify the device. The PPN is optional and defaults to the user's current account. The file specification is optional and defaults to *.*. You can use standard wildcards. If you specify a filename but not the type, BACKUP assumes a * type. If the filename ends with a period, BACKUP assumes a null type. If you specify a file type, you must also specify a filename.

output-backup-set-spec

Identifies the device where the Backup set is to be created. The Backup set specification should be a standard RSTS/E device specification of the format:

dev:[PPN]filename.typ

You must specify the device. PPN is optional and defaults to the user's current account. The file specification is optional and defaults to BACKUP.BCK. If the backup operation needs more than one volume in the Backup set, BACKUP prompts for subsequent volumes as follows:

```
Please dismount volume 1 of Backup set backup-set-name
Please mount volume 2 of Backup set backup-set-name
Where can this volume be located <default>?
```

Command Qualifiers

/[NO]ACCOUNT_DATA

If you specify /ACCOUNT_DATA and you have the necessary privileges (GACNT or WACNT), BACKUP preserves all pertinent information about the account (quotas, password, privileges, and so on). RSTS/E uses this data to recreate or update accounts during a RESTORE.

If you do not have accounting privileges (GACNT or WACNT), BACKUP only saves the attributes of accounts that you have access to. In particular, you can not access the password attribute or restricted user attributes.

If you specify /NOACCOUNT_DATA, RESTORE cannot create accounts that do not already exist on the output volume. The default is /ACCOUNT_DATA.

/BLOCK_SIZE=value

The /BLOCK_SIZE qualifier lets you select the block size used for the output Backup set. You must specify the value argument. Value can be in the range 2048 to 4080 bytes for tape, and 2048 to 7680 bytes for disk. Further, the value must be a multiple of 16 for tape and 512 for disk.

If you do not specify /BLOCK_SIZE, BACKUP uses 2048 bytes.

NOTE

> Use of this qualifier can affect the performance and reliability of BACKUP. If you use large values, BACKUP operates faster and can store more data into a volume. Smaller values cause less data to fit into a given volume; however, they can increase the reliability (see the description of the /GROUP_SIZE qualifier).

/BRIEF

Use this qualifier with /DIRECTORY to specify one file per line.

/BUFFER_SIZE=value

The /BUFFER_SIZE=value qualifier controls the size of the dynamic region used by BACKUP. The valid range is 3 to 127. The value represents the number of K-words of memory to be used by BACKUP as buffer space. In general, the large you specify n, the faster BACKUP performs. However, large numbers increase the system loading and resource usage, and can seriously impact other users.

If /BUFFER_SIZE=value is omitted, BACKUP uses the minimum buffer space consistent with the other qualifiers, or a value of 55 for MU: and MS: devices. If the amount of memory requested is not available, BACKUP uses as much memor as possible and displays an informational message. /BUFFER_SIZE=MAXIMUM uses as much available memory as possible without displaying any message.

/CREATED=(BEFORE=date-time, AFTER=date-time)
/CREATED=date-time
/MODIFIED=(BEFORE=date, AFTER=date)
/MODIFIED=date

The /CREATED and /MODIFIED qualifiers let you select the files to be backed up, as chosen by the file specification criteria (see the input-file-spec description), based on the creation date/time and/or revision date. If no file specification criteria exist, these qualifiers apply to all files on the input volume for this operation. You can use /CREATED and /MODIFIED in conjunction with each other; they are not mutually exclusive. To be selected, a file has to satisfy all the criteria you specify.

You can specify an exact date or range of dates for either qualifier. In the range format, each qualifier must take an argument of one or both of the subqualifiers BEFORE or AFTER. The subqualifier order in the argument is immaterial.

The date-time argument is of the standard form (for example, dd-mmm-yy or yy.mm.dd and hh:mm A/PM or hh:mm). You must specify the date. If you do not specify the time, it defaults to 00:00. In addition, if you specify the time, you must be separate it from the date by a colon (:).

The following generic terms can be used in the subqualifier:

- TODAY—meaning 00:00 on the current (system) date
- YESTERDAY—meaning 00:00 on the date previous to the current date

When used with /CREATED, the generic terms also accept a time argument.

If you do not specify either /CREATED or /MODIFIED, BACKUP selects all specified files for transfer to the Backup set.

NOTE

> Files transferred by BACKUP retain their original creation and revision dates (that is, the dates on the source volume). You need the DATES privilege to keep these original dates; otherwise, RSTS/E lists the current date as the creation date of the backed-up file.

/[NO]CRC

Controls the generation of 32-bit cyclic redundancy check (CRC) codes to better detect data errors. Every block header includes a 16-bit CRC, but this only detects errors in the header. The 32-bit CRC produced by the /CRC qualifier detects errors anywhere in the block. Generating CRC code blocks uses significant amounts of CPU time, and may degrade streaming performance on some systems.

When restoring a backup set written with /CRC, RESTORE automatically checks the CRC. If RESTORE finds a CRC error, it attempts to recover the data. The default is /NOCRC.

/DENSITY=n

Specifies the recording density on the output tape. The argument n can be any number between 2 and 32766, or the keywords MINIMUM or MAXIMUM. Generally, available tape drives support 800, 1600, or 6250 bpi.

If you specify MINIMUM or MAXIMUM, BACKUP checks the tape drive and sets the density to the minimum and maximum the drive supports. If you select a density that is not valid for the drive addressed, BACKUP displays a warning message and selects the highest density (below the specified value) that the drive supports. If you do not specify a density, the default value is the highest density available on the drive addressed.

You can only specify this qualifier for the first volume of a Backup set. Subsequent volumes are written at the same density as the first volume unless the addressed drive is different from the first drive and the original density is not available on the addressed drive. In this case, the density is the highest density available for the addressed drive.

/DIRECTORY

Tells BACKUP to list the files within a BACKUP saveset. When you use the /DIRECTORY qualifier, you can also use the qualifiers that go with the separate DIRECTORY command, such as /SELECT, /EXCLUDE, /FULL, and /BRIEF. /BRIEF is the default.

/END={REWIND | NOREWIND | DISMOUNT | NODISMOUNT}

/REWIND rewinds the tape before searching for the logical end of tape to create the backup file.

BACKUP must locate the logical end of tape so that it can create the backup file after any files already on the tape. While searching for the end of tape, BACKUI will check the name of each file it encounters and report an error if it finds a file with the same name as the backup file. This prevents having two files with the same name on the tape.

If you specify /REWIND, BACKUP rewinds the tape and scans the entire tape foı a duplicate file. If you specify /NOREWIND, then BACKUP scans from the tape's current position, and will not detect duplicate files located earlier on the tape. The default is /NOREWIND.

/EXCLUDE=(filespec list)

Lets you specify files that you want excluded from the BACKUP operation (for example, [0,1]SWAP?.SYS). You can include a list of file specifications in the command line up to the maximum length of a DCL command line.

/[NO]FILE_DATA

Controls the transfer of file data during a backup operation. If you specify /FILE_DATA, the operation proceeds with actual file data being transferred to the Backup set from the RSTS/E volume. If you specify /NOFILE_DATA, no files are written to the Backup set. Use /NOFILE_DATA with /ACCOUNT_DATA to back up only accounting data.

The default is /FILE_DATA.

/FULL

Use this qualifier with /DIRECTORY to specify an expanded directory of each file in the saveset. Several lines of information displays for each file.

/[NO]GROUP_SIZE[=n]

Specifies the group size for XOR redundancy. The optional argument can be a decimal number from 0 to 100. If you specify zero (equivalent to /NOGROUP_SIZE), BACKUP creates the Backup set without XOR redundancy. The default is /GROUP_SIZE=10.

The /GROUP_SIZE qualifier causes additional error recovery data to be stored in the Backup set. RSTS/E uses this information to recover from media errors during a RESTORE operation.

The smaller the number, the greater the chance of recovering from data errors on RESTORE. Note that a smaller number also makes the Backup set larger.

RESTORE can recover from one error in each group. So, for example, if you specify /GROUP_SIZE=20, RESTORE can recover from one error in every 20 blocks. That is, with /GROUP_SIZE=20, RESTORE cannot recover from 2 or more errors every 20 blocks. Note that the /BLOCK_SIZE qualifier sets the size of each Backup set block. So, if RESTORE cannot recover 2 blocks of data, you lose 2 blocks of that size. In general, if something cannot be recovered, you lose less data with a smaller /BLOCK_SIZE.

/INCLUDE=(filespec list)

Lets you specify files or accounts that you want to be unconditionally processed. You can include a list of file specifications in the command line up to the maximum length of a DCL command line.

Files in this set override those excluded with the /EXCLUDE qualifier.

/IN_LABEL=name

Specifies the name to be compared to the source volume ID. If you use this qualifier, the source volume ID must equal the specified name before the BACKUP can be perform the requested operation.

/[NO]INITIALIZE

Specifies the disposition of the output media prior to the data transfer operation and sets the default device specification in the "Where can this volume be located <default>?" prompt.

For disk volumes, /INITIALIZE specifies that any file structure on the disk is to be disregarded and the volume is to be reinitialized with a RSTS/E file structure as if it were done with the INITIALIZE command.

BACKUP also creates a nonuser account on the disk to hold the Backup set. That account is the account explicitly specified in the output specification, or implicitly specified as the current account of the user issuing the command. The pack cluster size is 16, the UFD cluster size is 16, and the file cluster size for the save set is set to the amount of free space on the disk divided by 100, rounded up to the next highest power of two. BACKUP does not define any accounting data for the created account.

For magnetic tape volumes, /INITIALIZE specifies that BACKUP initializes the tape as a standard ANSI tape.

You can only specify /NOINITIALIZE when doing a BACKUP to tape to create a new Backup set to be appended to the end of a tape containing other files (possibly other Backup sets).

/[NO]LIST_FILE[=filespec]

Controls the production of a directory-like listing file that catalogs the sequence of events in the BACKUP operation. The file specification can be any valid RSTS/E file specification (wildcards are not allowed). The default is your terminal.

You can use this qualifier to produce a listing of files in the Backup set.

/OUT_LABEL=name

Specifies the desired name for the output volume ID. The resultant name can vary depending on the interaction between /OUT_LABEL and /[NO]INITIALIZE.

Table 16–1 summarizes the possible outcomes of the interaction of /[NO]INITIALIZE and /OUT_LABEL during BACKUP.

Table 16–1: /INITIALIZE and /OUT_LABEL Interaction During BACKUP

Qualifiers Specified	Output Volume ID
/OUT_LABEL=name /INITIALIZE	The specified name.
/OUT_LABEL=name /NOINITIALIZE	The specified name; however, the operation is only successful if the output volume label has the same name.
/INITIALIZE (only)	BACKUP supplies the name.
/NOINITIALIZE (only)	The name is unchanged.

/[NO]OUTPUT[=filespec]

The same as the /LIST_FILE qualifier.

/[NO]PROMPT

Normally, the Backup utility issues a MOUNT prompt, for each BACKUP operation even if the Backup set's device is already mounted. You can eliminate the prompt for the first volume by including the /NOPROMPT qualifier in your command line. The default is /PROMPT.

/[NO]QUERY

Causes BACKUP to prompt the user for each file that meets the selection criteri as to whether or not the file should be transferred. A Y[ES] response effects the transfer. A response of N[O] or pressing Return bypasses the file. Any other response causes the prompt to be reissued. The default is /NOQUERY.

/[NO]REWIND

For tape backup files, indicates whether to rewind the tape before searching for the logical end of tape to create the backup file.

BACKUP must locate the logical end of tape so that it can create the backup file after any files already on the tape. While searching for the end of tape, BACKUI will check the name of each file it encounters and report an error if it finds a file with the same name as the backup file. This prevents having two files with the same name on the tape.

If you specify /REWIND, BACKUP rewinds the tape and scans the entire tape fo a duplicate file.

If you specify /NOREWIND, then BACKUP scans from the tape's current position and will not detect duplicate files located earlier on the tape. The default is /NOREWIND.

/SELECT[=(filespec list)]

Specifies the files or accounts to restore from the backup file. Use this qualifier t restore a single file or a set of files from the backup file.

If you don't include this qualfier, then all files and accounts in the backup file are restored, unless you specify other selection qualifiers.

/[NO]VERIFY

The /VERIFY qualifier specifies that a comparison operation is to be performed on all data transferred. This post-process verification reports the presence of any data differences. For any differences found, BACKUP displays the PPN, file name, and file type at the user's terminal. The default is /NOVERIFY.

Examples

Table 16–2 shows command formats for BACKUP operations and some of the qualifiers associated with a backup operation.

Table 16–2: BACKUP Quick Reference Table

Command Action	Command Example
Save a file to a save set on tape	BACKUP DR1:PAYROL.DAT MT0:PAYROL.BCK
Save a disk to a save set on tape	BACKUP/ACCOUNT_DATA DU1:[*,*]*.*- MM0:1409MA.BCK
Save a list of files to a save set on tape	BACKUP DR2:PAYROL.DAT,DR2:BENEFI.DAT,- DR2:VACATI.DAT MT0:DAT17M.BCK
Full backup of a disk	BACKUP/BUFFER_SIZE=75/INITIALIZE- /LIST_FILE=SYDISK [*.*]*.* MU0:/END=REWIND- /EXCLUDE=[0,1]??????.SYS/NOPROMPT/VERIFY
Incremental backup of a disk (following earlier full backup)	BACKUP/BUFFER_SIZE=75/INITIALIZE- /CREATE=AFTER=4-FEB-90:9:28- /LIST_FILE=SYDISK [*.*]*.* MU0:/END=REWIND- /EXCLUDE=[0,1]??????.SYS/NOPROMPT/VERIFY

16.2.2 RESTORE Command

The RESTORE command restores a RSTS/E disk or subset thereof from a tape or disk Backup set. Once RSTS/E determines the syntactic validity of the command specification and issues any warning message relative to either the command or the environment, RESTORE prompts with:

```
Please mount volume 1 of Backup set backup-set-name Where can this
volume be located <default>?
```

At the prompt, enter the device to use or press Return or Line Feed to accept the default value. The default value is the device specified as the input parameter. If you specify /NOPROMPT, RESTORE omits the prompt message.

If you are doing a partial restore, and you know what volume to start with, start with that volume (although you may start with any volume). RESTORE issues a confirmation message and starts with the volume you specify. Note that RESTORE may issue a warning message in this case.

NOTE

> If you have sufficient privilege and you specified /ACCOUNT_DATA in the RESTORE command, and if the Backup set contains accounting information, BACKUP creates accounts if they did not previously exist. The account characteristics are the same as they were for the Backup set accounts. If the account already exists, BACKUP replaces the existing account characteristics.
>
> Use RECOVR.COM or the latest installation kit to reboot the system before you restore system disks.

Format

RESTORE *input-backup-set-spec output-file-spec*

Command Qualifiers	Defaults
/[NO]ACCOUNT_DATA	/ACCOUNT_DATA
/BRIEF	none
/BUFFER_SIZE=n	See Discussion
/CREATED=(date/time or range)	ALL if neither /CREATED
/MODIFIED=(date or range)	or /MODIFIED is present
/DIRECTORY	none
/END=[NO]REWIND [,[NO]DISMOUNT]	/END=DISMOUNT
/EXCLUDE=(filespec list)	None excluded
/[NO]FILE_DATA	/FILE_DATA
/FULL	none
/INCLUDE=(filespec list)	None included
/IN_LABEL=name	Volume ID is not checked
/[NO]INITIALIZE	See Discussion
/[NO]LIST_FILE[=filespec]	/NOLIST_FILE
/OCCURRENCE=number	/OCCURRENCE=1
/[NO]OPTIMIZE[=(list)]	See Discussion
/OUT_LABEL=name	See Discussion
/[NO]OUTPUT[=filespec]	/NOOUTPUT
/PLACED_POSITION[=keyword]	/PLACED_POSITION=INDEX

/[NO]PROMPT	/PROMPT
/[NO]QUERY	/NOQUERY
/[NO]REPLACE[=QUERY]	/REPLACE=QUERY
/[NO]REWIND	/NOREWIND
/SELECT=(filespec list)	See Discussion
/[NO]VERIFY	/NOVERIFY

Prompts

See Discussion

Command Parameters

input-backup-set-spec

Identifies the device where the first volume of the Backup set is mounted. The Backup set specification should be a standard RSTS/E device specification of the format:

dev:[PPN]file-spec

You must specify the device. PPN is optional and defaults to the user's current account. File-spec is optional and defaults to BACKUP.BCK. If the Backup set has more than one volume, RESTORE prompts you for subsequent volumes as follows:

```
Please dismount volume 1 of Backup set backup-set-name
Please mount volume 2 of Backup set backup-set-name
Where can this volume be located <default>?
```

output-file-spec

Identifies where to put the files or accounts to be restored. The file specification should be a standard RSTS/E file specification of the format:

dev:[PPN]file-spec

You must specify the device. PPN is optional and defaults to the account you logged in to. File-spec is optional and defaults to the file specification(s) recorded on the Backup set.

If you specify a PPN, RESTORE transfers all selected files to that account, even if the Backup set has multiple accounts, assuming you have sufficient privilege to create files in that account (GWRITE or WWRITE). If file name conflicts occur as a result of the RESTORE (for example, TEMP.BAS exists in two or more accounts on the Backup set), the second and subsequent transfers require operator intervention unless you specify the /REPLACE qualifier.

If you specify a wildcard PPN, RESTORE transfers all selected files to:

- [*,*]—the same accounts as on the Backup set
- [n,*]—forces the project number to n, uses the programmer number as on the Backup set
- [*,n]—forces the programmer number to n, uses the project number as on the Backup set

If the accounts do not exist and you specify /ACCOUNT_DATA, RESTORE creates them with the same characteristics as they have on the Backup set, providing you have the necessary privilege to create the accounts (GACNT or WACNT).

Command Qualifiers

/[NO]ACCOUNT_DATA

If you have sufficient privilege (GACNT or WACNT), this qualifier determines what level of information from a UFD is restored from a Backup set.

Use /ACCOUNT_DATA to restore all accounting data that is recorded in the Backup set both to existing and to newly created accounts. Use /NOACCOUNT_DATA to inhibit the creation of accounts and restoration of UFD data. The default is /ACCOUNT_DATA.

NOTE

> If an account already exists, RESTORE supersedes the accounting data when restoring to the original account.

/BRIEF

Use this qualifier with /DIRECTORY to specify one file per line.

/BUFFER_SIZE=value

The /BUFFER_SIZE=value qualifier controls the size of the dynamic region that RESTORE uses. The valid range is 3 to 127. The value represents the number of K-words of memory that RESTORE can use as buffer space. If you omit /BUFFER_SIZE=value, RESTORE uses the minimum buffer space consistent with the other qualifiers.

The larger value you specify, the faster RESTORE performs. However, note that large values cause a greater impact to other users.

If the amount of memory you request is not available, RESTORE uses as much memory as possible and displays an informational message. If you specify /BUFFER_SIZE=MAXIMUM, RESTORE uses as much memory as possible; however, it does not display any message.

/CREATED=(BEFORE=date-time, AFTER=date-time)
/CREATED=date-time
/MODIFIED=(BEFORE=date, AFTER=date)
/MODIFIED=date

The /CREATED and /MODIFIED qualifiers let you select the files to be restored, as chosen by the file selection criteria (see the output-file-spec description), based on the creation dates/times and/or revision dates. If no file selection criteria exist, these qualifiers apply to all files on the input volume to which you have access for this operation. You can use /CREATED and /MODIFIED in conjunction with each other; they are not mutually exclusive. To be selected, a file has to satisfy all the criteria you specify.

You can specify an exact date or a range of dates for either qualifier. In the range of dates format, each qualifier must take an argument of one or both of the subqualifiers, BEFORE or AFTER. The order of the subqualifiers in the argument is immaterial. If you specify both BEFORE and AFTER dates for either one of the qualifiers, the AFTER date must be earlier than the BEFORE date.

If you do not use either /CREATED or /MODIFIED, all specified files are selected for transfer from the Backup set.

NOTE

> BACKUP retains both the creation and revision dates as they were on the source volume for the files transferred if you have DATES privilege.

/DIRECTORY

Tells BACKUP to list the files within a BACKUP saveset. When you use the /DIRECTORY qualifier, you can also use the qualifiers that go with the separate DIRECTORY command, such as /SELECT, /EXCLUDE, /FULL, and /BRIEF. /BRIEF is the default.

/END={REWIND | NOREWIND | DISMOUNT | NODISMOUNT}

(For magnetic tapes only) REWIND causes the magnetic tape to be rewound to the load point and dismounted following successful completion of the RESTORE operation. NOREWIND causes the magnetic tape to remain positioned after the most recent Backup set. The default is NOREWIND.

If you want the Backup set's device to remain mounted at the end of the BACKUP operation, specify NODISMOUNT in the command line. The default is DISMOUNT.

/EXCLUDE=(filespec list)

Lets you specify files that you want to be excluded from the RESTORE operation. You can include a list of file specifications in the command line up to the maximum length of a DCL command line.

/[NO]FILE_DATA

Controls whether file data is transferred during a RESTORE operation. If you specify /FILE_DATA, RESTORE transfers the file data from the Backup set to the RSTS/E volume. If you specify /NOFILE_DATA, RESTORE proceeds without restoring any of the specified files. The default is /FILE_DATA.

If you have accounting (GACNT, WACNT) privileges, you can use /NOFILE_DATA with the /ACCOUNT_DATA qualifier to restore only account information from a Backup set.

/FULL

Use this qualifier with /DIRECTORY to specify an expanded directory of each file in the saveset. Several lines of information displays for each file.

/INCLUDE=(filespec list)

Lets you specify files that you want to be unconditionally processed. You can include a list of file specifications in the command line up to the maximum length of a DCL command line.

Files in this set override those specified with the /EXCLUDE qualifier.

/IN_LABEL=name

Specifies the name to be compared to the source volume ID. If you use this qualifier, the source volume ID must equal the specified name before the RESTORE operation can be performed.

/[NO]INITIALIZE

Specifies the disposition of the output media prior to the data transfer operation.

The /INITIALIZE qualifier specifies that RESTORE should disregard any file structure on the disk and reinitialize the volume with a RSTS/E file structure as if it were done with the INITIALIZE command.

RESTORE takes the characteristics from the source device summary. (If the device cluster size of the original volume is too small, the next higher valid cluster size is used for the output volume.)

You can only specify /NOINITIALIZE when the RESTORE operation is from a Backup set to an existing RSTS/E disk. In this case, RESTORE places the files on the existing file structures instead of replacing them.

/[NO]LIST_FILE[=file-spec]

Generates a directory-like listing file, cataloging the sequence of events in the RESTORE operation. If you specify /LIST_FILE, you must include the file specification. It can be any legal RSTS/E file specification (wildcards are not allowed). The default is /LIST_FILE=KB:.

/OCCURRENCE=number

Indicates the particular occurrence of a named Backup set to use when more tha one Backup set of the same name exists on a magnetic tape. You must specify th argument which refers to the position relative to either the current tape positior or the beginning of the tape depending on the use of /[NO]REWIND.

RSTS/E Backup sets are interchangeable with VAX Backup sets. However, VAX Backup set names can be greater than six characters, while RSTS/E looks at only the first six characters. Therefore, RSTS/E cannot distinguish between two VAX names that have the same first six characters (for example, ABCDEF and ABCDEFG). To transfer ABCDEFG, you must use ABCDEF/OCCURRENCE=2.

The default is /OCCURRENCE=1; the first occurrence of the named Backup set.

/[NO]OPTIMIZE[=(list)]

The /[NO]OPTIMIZE=(list) qualifier controls RESTORE's optimization of the directory structure. Two optimizations are available:

- Directories can be extended to the size they were on the original volume
- File cluster sizes can be increased to the maximum necessary to minimize calls to the file processor

The /NOOPTIMIZE qualifier inhibits these optimizations. The /OPTIMIZE qualifier selects all optimizations. The default is /OPTIMIZE.

The /OPTIMIZE=(list) qualifier selects optimizations from the following:

- CLUSTERSIZE—Optimize cluster size
- DIRECTORY—Optimize directory size

In general, you would want to perform these optimizations to increase overall system performance. However, if there is not much contiguous disk space available, you would not want to make all executable files contiguous. Similarly, if there is little free disk space, you would not want to optimize cluster sizes because this takes more disk space.

/OUT_LABEL=name

Specifies the desired name for the output volume ID. The resultant name can vary depending on the interaction between /OUT_LABEL and /[NO]INITIALIZE.

Table 16–3 summarizes the possible outcomes from the interaction of /[NO]INITIALIZE and /OUT_LABEL during RESTORE.

Table 16–3: /INITIALIZE and /OUT_LABEL Interaction During RESTORE

Qualifiers Specified	Output Volume ID
/OUT_LABEL=name /INITIALIZE	The specified name.
/OUT_LABEL=name /NOINITIALIZE	The specified name; however, the operation is only successful if the output volume label has the same name.
/INITIALIZE (only)	The destination volume name is the same as the source volume name.
/NOINITIALIZE (only)	The name is unchanged.

/[NO]OUTPUT[=filespec]

The same as the /LIST_FILE qualifier.

/PLACED_POSITION[=keyword]

Controls the placement of files that were marked as placed on the source disk.

The /PLACED_POSITION qualifier takes one of the following keyword arguments:

- INDEX—Place close to the storage allocation table and master file directory
- MIDDLE—Place close to the middle of the disk
- ORIGINAL—Place close to the original position on the source disk.

The default is INDEX

The rules for contiguity apply first and foremost; that is, RESTORE sacrifices placement in the interest of contiguity. The first sufficient contiguous space closest to the selected position is the first choice for positioning the file on the output volume.

If no contiguous space of sufficient size exists on the output volume, then the first cluster(s) of the file are positioned as close to the original address as possible and the remaining clusters are positioned as required.

/[NO]PROMPT

Normally, the BACKUP utility issues a MOUNT prompt for each RESTORE operation even if the Backup set's device is already mounted. You can eliminate the prompt for the first volume by including the /NOPROMPT qualifier in your command line. The default is /PROMPT.

/[NO]QUERY

Causes RESTORE to prompt you as to whether each file that meets the selection criteria should be transferred. A Y[ES] response effects the transfer. A response of N[O] or pressing Return bypasses the file. Any other response causes RESTORE to reissue the prompt. The default is /NOQUERY.

/[NO]REPLACE[=QUERY]

The /REPLACE qualifier specifies that if a file being transferred in a restore operation already exists on the destination disk, the file from the Backup set replaces the one already on the disk. The /REPLACE=QUERY qualifier tells RESTORE to query the user before replacing a file.

The /NOREPLACE qualifier specifies that if a name conflict occurs, the file transfer is not to take place under any circumstances.

The default is /REPLACE=QUERY.

/[NO]REWIND

Specifies the disposition of a magnetic tape Backup set volume before processing. The default is /NOREWIND.

If you specify /REWIND, the search for the Backup set begins at the beginning of the tape. The /NOREWIND qualifier means the search starts at the present position.

If the logical end-of-tape on the volume is found before the desired Backup set, the tape is rewound and an error message appears at the terminal.

/SELECT[=(filespec list)]

Lets you select specific files from the input Backup set to be restored. You can use the /SELECT qualifier in conjunction with the /EXCLUDE qualifier and/or the /INCLUDE qualifier. If you use /SELECT in combination with other qualifiers, the file transfer candidates are chosen according to the following steps:

1. All files on the volume for which the user has read access privilege are initially considered
2. If you use the /CREATED or /MODIFIED qualifiers, only those files from step 1 which meet the time bracket criteria are considered
3. If you use the /SELECT qualifier, only those files from step 2 that match one of the /SELECT file specifications are considered
4. If you use the /EXCLUDE qualifier, any files from step 3 that match one of the /EXCLUDE file specifications are removed from consideration
5. If you use the /INCLUDE qualifier, RESTORE also considers any files meeting the inclusion criteria, provided that you have access to them. Note that RSTS/E does not consider the /CREATED, /MODIFIED, /SELECT, or /EXCLUDE qualifiers when processing the /INCLUDE list.

/[NO]VERIFY

The /VERIFY qualifier specifies that a comparison operation is to be performed on all data transferred. This post-process verification (for multi-volume Backup sets, all volumes are transferred before any volume is verified) reports the presence of any data differences. For any differences found, RESTORE displays the PPN, file name, and file type at the user's terminal. The default is /NOVERIFY.

Examples

Table 16–4 shows command formats for RESTORE operations and some of the qualifiers associated with restore operations.

Table 16–4: RESTORE Quick Reference Table

Command Action	**Command Example**
Restore a file from a save set on tape	RESTORE/SELECT=[*,*]PAYROL.DAT - MT0:PAYROL.BCK DR1:*.*
Restore a disk from a save set on tape	RESTORE MM0:1409MA.BCK DU1:[*,*]*.*
Restore a list of files from a save set on tape	RESTORE MT0:DAT17M.BCK/SELECT=- (PAYROL.DAT,BENEFI.DAT,VACATI.DAT) DR2:[*.*]*.*
Restore a disk from a save set, if file exists, replace it.	RESTORE/LIST_FILE=928FEB.LOG- /ACCOUNT_DATA MM0:928FEB.BCK DR1:[*.*]*.*

Appendix A

Managing the OPSER Spooling Package

This appendix describes the OPSER-based spooling package. This package, like the new Print/Batch Services (PBS) package (see Chapter 9), provides print and batch services. Most of the programs require SWCFG privilege to run. General aspects of related user programs, such as QUE and BATCH, are mentioned only briefly in this appendix. See the *RSTS/E Utilities Reference Manual* for a full description.

The information presented here is for backwards compatibility only. New applications should use the PBS package. Move existing applications to PBS over time. PBS is:

- More efficient
- More flexible
- Easier to manage
- Handles DCL command files
- Permits 8-bit and control characters on printers
- Has job limits

A.1 Overview of Operator Services

Operator services on RSTS/E involve the OPSER program and controlled programs in OPSER tables. Controlled (online) programs are:

- QUEMAN
- SPOOL
- BATCH
- RESTOR

SPOOL and BATCH are spooling programs executing queued requests for either a line printer device or batch processor. QUEMAN is the queue manager program that passes queued requests to spooling programs and adds, updates, and deletes requests in the various queues. RESTOR is the optional system data restoring package that OPSER can control.

A.1.1 OPSER Program Overview

The OPSER program establishes interjob communication for all of its controlled programs. OPSER declares itself a message receiver with a system-wide identification that the controlled programs can recognize. When a controlled program starts, it declares itself a message receiver and supplies OPSER with certain data by means of the system message send/receive SYS call. OPSER places the program in its table of online jobs. See the *RSTS/E Programming Manual* for information about SYS calls.

The identification that OPSER uses to communicate with online programs is called the message receiver identification, or simply, the receiver identification. Each identification exists in the system message receiver table and must be unique. Because of this uniqueness, only one copy of OPSER can be running on your system. Additionally, because all control and interjob communication of spooling jobs depends on OPSER, the operator services program must be running before any controlled jobs can be run.

After it establishes initial interjob communication, OPSER makes it possible for an operator to interact with the controlled jobs. OPSER becomes the main interface between the operator and a system controlled program. Figure A–1 shows the interaction among system controlled programs and the operator. OPSER broadcasts information to the operator through a terminal designated as the Operator Services Console (OSC). For flexibility, the operator can make any keyboard line on the system the the OSC. To isolate the keyboard control from unwarranted tampering, OPSER itself does not need to be connected to the OSC, but may broadcast data on the physical keyboard line. Given that intermittent output is generated on the OSC, any other user (including the operator) could be logged into the system on that terminal.

OPSER itself establishes a data base by which it controls online jobs. The data base resides in the OPSER package account OPSER$: and consists of three work files:

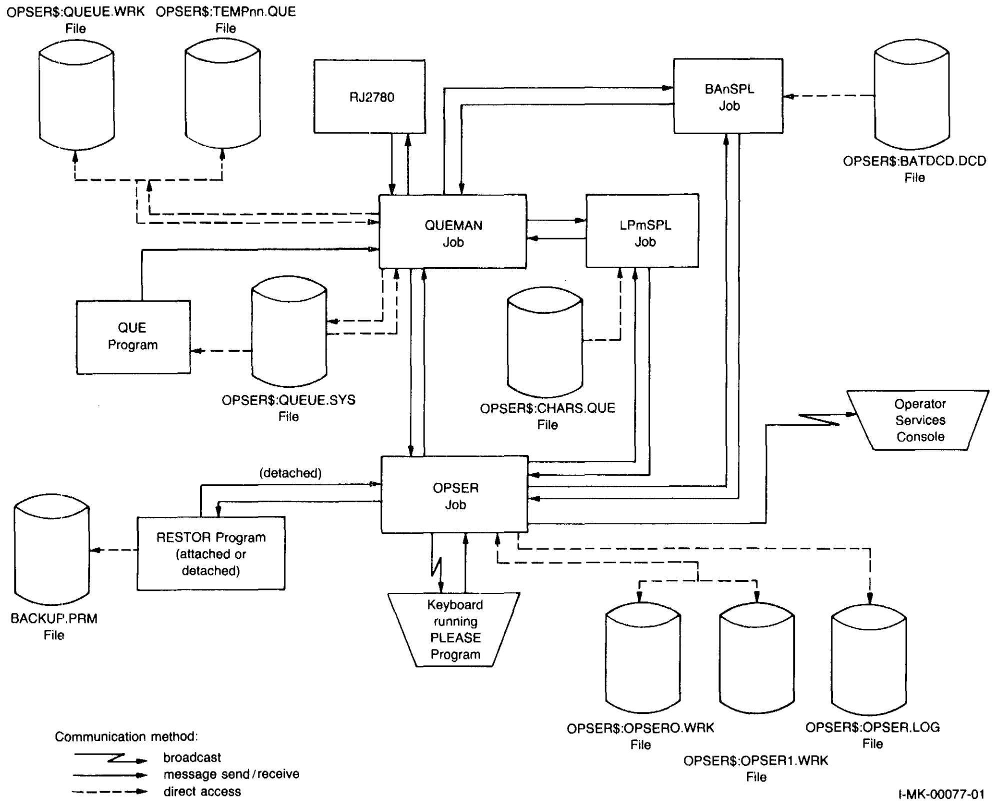
- OPSER0.WRK—Contains the current activities and message control directory tables
- OPSER1.WRK—Has the table of jobs on line to OPSER, the valid operator table, and tables of messages and legal commands
- OPSER.LOG (which can be optionally renamed)--Provides a history of operating activity

To provide a nonvolatile data base during a time-sharing session, as well as continuity of operations between time-sharing sessions, OPSER stores the work file information on disk. Thus, if the system crashes or if OPSER alone unexpectedly terminates, the work files retain the most current processing information. Similarly, these files maintain data continuity from one time-sharing session to another.

Whenever OPSER starts, it examines its files to determine whether entries in the online job and valid operator tables are still valid. Generally, OPSER retains active and valid entries and purges any entries that do not satisfy validation requirements. Upon restarting, therefore, OPSER attempts to preserve data from its previous operational state.

The most convenient way to interact with OPSER is through the PLEASE program. PLEASE transmits commands to the OPSER program. OPSER checks that the sender is a valid operator and, if so, takes the requested action. Becaus of PLEASE, the operator need never directly run or attach to OPSER to perform operator functions on controlled jobs. See the section "Operator Communication Program: PLEASE" for a description of the PLEASE program.

Figure A–1: System Controlled Programs and Operator Interaction



OPSER provides processing checks for online jobs. OPSER makes sure that all online jobs are still active, are still valid receivers, and are not hibernating. If any job is found to be hibernating, OPSER notifies the operator, who can attach the job to a terminal, remedy the cause of the hibernation, and/or restart the job.

OPSER interaction between a controlled job and the operator is recorded in one of three ways:

- As a message—A message is specifically formatted data to which OPSER assigns a sequence number for operator reference.
- As an action request—An action request is a special message that requires operator action and a response before a controlled job can resume processing. A request usually involves both performing an action (such as mounting a volume) and typing an answer (such as a device designator or program interrupt command) in direct response to the request.
- As an information line—An information line is a free form report of some internal operation performed. Information lines are typically not important to the operator function but merely provide a history of online events.

A.1.2 QUEMAN Program Overview

The QUEMAN program manages the system queue file OPSER$:QUEUE.SYS. The file retains all pending user requests and can store requests between time-sharing sessions. Users make requests of spooling programs through the QUE system program. To form a request for a user, QUE builds a message and sends it to QUEMAN, which updates the QUEUE.SYS file.

QUEMAN maintains a table of online spooling programs in a work file, QUEUE.WRK, which is stored in the OPSER package library (OPSER$:). The program passes a queued request to a destination spooling job not currently busy with a request. As a spooling job completes a request, it notifies QUEMAN, which updates the queue and work files and checks for another request to pass to the spooling job.

A spooling program typically handles one request at a time from QUEMAN. If no request is pending for a spooling program, the job enters an indefinite sleep state. The job is not awakened until the system queues a message for it. The message is the result of QUEMAN or OPSER transmitting data to the spooling job by means of the system message send/receive mechanism.

To retain as much data as possible between time-sharing sessions, QUEMAN, like OPSER, examines its files upon restarting. After restarting, QUEMAN should resume processing with no loss of data from its previous run. QUEMAN examines the queue file structure and contents and reestablishes synchronization with any spooling jobs already on line and active.

A.1.3 SPOOL Program Overview

The SPOOL program handles requests made for line printer output. SPOOL maintains dual communication paths: one with QUEMAN and one with OPSER. QUEMAN transmits queued requests to SPOOL. SPOOL interacts with QUEMAN only to notify it that a transmitted request is completed. Under normal circumstances, communication between SPOOL and OPSER is necessary only when an operator requires some special action (for example, aligning forms) of the spooling job.

To identify output of a user's queued request, SPOOL prints heading burst pages. The program accesses the character generation file OPSER$:CHARS.QUE to form the large, easily readable letters necessary to distinguish discrete requests and files within a request.

A.1.4 BATCH Program Overview

The BATCH program executes user requests by running a job for a user on a static pseudo keyboard defined with the SET SYSTEM command. By using a pseudo keyboard, BATCH eliminates the requirement for a physical terminal and relieves the user from typing by processing commands from a disk file.

BATCH, like SPOOL, maintains dual communication paths: one with QUEMAN and one with OPSER. QUEMAN sends a message to BATCH to process a job request. BATCH interacts with QUEMAN only to notify it that a transmitted request is completed. BATCH typically depends on OPSER to form requests for operator action. To mount a user volume, for example, BATCH notifies OPSER. In turn, OPSER generates the request for the operator and processes the response from the operator.

To process commands from a user-queued disk file, BATCH requires the command decoding file OPSER$:BATCH.DCD. The BATCH.DCD file contains all the codes and parameters BATCH needs to decode user-specified control statements, perform syntax checking, and form a special intermediate file (BAnJmm.COM) of encoded BATCH commands.

Both the SPOOL and BATCH programs charge processing (execution) time to the account under which the user queues the request. Overhead time, used to set up a request, is charged to the account under which the program is running.

A.1.5 RJ2780 Program Overview

The RJ2780 program is an optional layered product, available through DECUS. Unlike BATCH and SPOOL, RJ2780 maintains only one communication path. QUEMAN sends a message to RJ2780 to process the next request (send a file). The RJ2780 program can also send a message to QUEMAN requesting that SPOOL print a file that RJ2780 has received. RJ2780 has no communication with OPSER.

A.1.6 Controlling RESTOR with OPSER

An operator can control the RESTOR program through OPSER. If RESTOR runs detached, control through OPSER is the only means by which the operator can communicate with the RESTOR job. RESTOR, when detached, transmits all operator requests through OPSER and receives responses only from OPSER.

A.1.7 Overview of OPSER Shutdown

The entire operator services package is designed to terminate in a controlled manner using a "shutdown level." The defined shutdown level for each job determines the order in which OPSER terminates controlled jobs during system shutdown. The SHUTUP system program can be run so OPSER terminates controlled jobs with the lowest shutdown level first. OPSER informs each job at a certain level to complete processing at a logical end point. Only after OPSER shuts down all jobs at a given level does it proceed to shut down all jobs at the next highest level. SHUTUP itself does not proceed with final system shutdown until OPSER terminates all of its controlled jobs and kills itself. By using the controlled shutdown of OPSER, you can ensure continuity of processing user requests from one time-sharing session to another.

Since RJ2780 does not communicate with OPSER, OPSER does not automatically shutdown RJ2780 as it does SPOOL and BATCH.

A.2 Operator Services Program: OPSER

The operator services program OPSER consists of two modules:

- OPSER—Establishes the initial conditions on startup, checks initial conditions upon restart, and chains to the second module, OPRUN
- OPRUN—Executes commands and periodically checks status.

Each module is stored in the OPSER Package Library (OPSER$:), has a protection code of <232>, and requires SWCFG privilege to run. For simplicity, this appendix refers only to one program, OPSER, which functionally includes the two modules.

You must run the OPSER program before the queue manager and spooling programs. If your applications need to use the OPSER package, you should start OPSER by including the RUN command in your system startup command file. To start OPSER, type the following command:

```
$ RUN OPSER$:OPSER
OPSER V9  RSTS V9  TIMESHARING

#
```

At this point, the OPSER program performs a number of checks before it prints the number sign (#) prompt. The remaining portion of this section describes these initial program checks. Once you are familiar with this procedure, you can learn about OPSER operator commands, described in the following sections.

If the you have SWCFG privilege, the program prints an identification line at the terminal to indicate it is running. If you do not have SWCFG privilege, OPSER displays an error message to indicate that you do not have access to OPSER.

OPSER tests for the presence of its two work files (OPSER0.WRK and OPSER1.WRK) in OPSER$:. If the work files are not present, OPSER prints the warning message:

```
%OPSER files not found - will initialize ...
```

OPSER then initializes all its tables. At this point, the valid operator table has a single entry that allows the current user (on any terminal) to communicate through OPSER. If ERRCPY is active, the online job table has ERRCPY as its only entry.

When OPSER finds the work files, it makes sure that it has write access to the files. Should another job have write access to the files, OPSER prints the following warning message and terminates:

```
%OPSER does not have write privileges to its files
```

To allow OPSER to gain write access, the operator must terminate the job currently having write access and run OPSER again.

The program performs further special processing for currently existing work files. The processing makes sure of data integrity in case OPSER is being restarted after a system crash. A check is made of the entries in the online job and valid operator tables.

For each entry in the online job table, OPSER checks the following conditions:

- The job number indicated in the entry must be active on the system.
- The receiver identification in the entry must exist in the system message receiver table for that job number.
- The account in the entry must be a valid account number.

An entry for an inactive or otherwise illegitimate job is removed from the online job table. For each active and legitimate job, OPSER requests retransmission of the job's last message when processing begins. When processing begins, OPSER generates a command to display the online job table.

OPSER checks the entries in the valid operator table. Additional actions take place depending on whether jobs found in the online job table are active. If the online job table has no active jobs and a meaningless entry is in the valid operator table, OPSER clears the table and sets initial conditions. The resulting operator table has one entry that allows, as valid operators, the current user from any keyboard. If the online job table has an active job, the program merely removes those entries that have no meaning from the valid operator table. To signal a change in the current table, OPSER creates an internal command to print the valid operator list when processing begins.

After processing of the work files is complete, the program attempts to declare OPSER as a receiving job. If the declaration fails, the program prints the error message:

```
?OPSER cannot declare itself a receiver
```

This message indicates that another copy of OPSER has been run and must be properly terminated to allow the current copy to run successfully.

OPSER establishes the current keyboard as the Operator Services Console (OSC) and a file named OPSER.LOG as the default log file. OPSER.LOG is optional, but OPSER uses it until an explicit command closes it or assigns another log file. When OPSER begins normal operation, it executes any internally generated commands to list online jobs and valid operators. After executing the commands, OPSER is ready to accept data sent to it by any program. OPSER prints the number sign (#) character to show that it is ready to accept commands.

A.2.1 OPSER Operator Commands

You can abbreviate a valid OPSER command to three or more characters and separate the command from any operands by a space character. No embedded spaces are allowed within a command. Delimiters within the text of the operands are the semicolon (;) and colon (:) characters and must be present to delimit elements of text. Throughout the following discussion, note that square brackets enclose optional items.

Table A–1 summarizes the commands and formats. Note that the circumflex (^) character marks the location of a required space.

Table A–1: OPSER Commands

Command Name	Syntax and Meaning
ANSWER	ANS[WER]^msgnumber:text Conveys the text following the colon as a response to an action request denoted by message number. Also deletes the action request.
CHANGECONSOLE	CHA[NGECONSOLE]^KBn: Changes the operator services console to the keyboard unit designated by n. The unit should be on line but need not be free. It must also be set for /BROADCAST (see Chapter 8).
DELETE	DEL[ETE]^msgnumber Deletes from an OPSER internal table an unanswered action request denoted by its message number. DEL[ETE]^#m[:n] Deletes the n oldest (from 1 to 32) action requests for a given job. The command requires the number sign (#), the number (m) of the sending job, a colon (:), and the number (n) of requests to delete.
DETACH	DET[ACH] Detaches the OPSER program from its keyboard, after which the operator can run PLEASE to communicate with OPSER.
EXIT	EXI[T] Closes the log file and work files, removes OPSER from the system message receiver table, and terminates the program. If the job is attached, the program returns you to the keyboard monitor prompt. If detached, the program broadcasts a message to the system console terminal (KB0:) and kills itself.
INTERRUPT	INT[ERRUPT]^rcvrid:text INT[ERRUPT]^#n:text Sends the unsolicited text to an on-line job specified either by its message receiver identification or by #n (where n is the number of the job under which the on-line program is running). The LIST JOBS command listing gives job numbers and receiver identifications.
LIST	LIS[T]^JO[BS] LIS[T]^OP[ERATORS] Prints a listing of all jobs on line to OPSER or a listing of valid operator accounts and keyboards.

(continued on next page)

Table A–1 (Cont.): OPSER Commands

Command Name	Syntax and Meaning
LOGFILE	LOG[FILE]^[file];[msglevel] Establishes, as the OPSER log file, the specified file or devic and sets the message level to the value specified. If a file or device is not specified, a semicolon followed by a level simply changes the message level of the open log file. If neither a file nor message level is specified, OPSER closes the current log file and records nothing more.
	LOG[FILE]^[file];AL[L] ALL records all messages and action requests.
	LOG[FILE]^[file];RE[QUESTS] REQUESTS records only action requests.
	LOG[FILE]^[file];NO[NE] NONE records no messages, action requests, or OPSER information lines.
MESSAGE	MES[SAGE]^[level] Sets the level of messages for the OSC. The level controls what types of messages are printed at the OSC. If no level is included in the command, an error is generated. MES[SAGE]^AL[L] ALL includes all messages, action requests, and OPSER information lines.
	MES[SAGE]^RE[QUESTS] REQUESTS includes only action requests and information lines and excludes messages.
	MES[SAGE]^NO[NE] NONE means that nothing is printed on the OSC. (See the section "Message Types" for a discussion of message level.)
OPERATOR	OPE[RATOR]^KBn:[p,pn] Updates the valid operator list with the designated keyboard unit and account combination. The asterisk (*) character can replace the keyboard unit, project number(p), and programmer number (pn) to include all of that element.
	OPE[RATOR]^-KBn:[p,pn] If a minus (-) sign precedes the specification, that specific element is deleted from the list.
RETYPE	RET[YPE]^msgnumber Reprints, at the requesting terminal, an unanswered action request denoted by its associated message number.
	RET[YPE]^[#n[:n]] Several unanswered action requests can be printed by giving the number of the sending job. If :n is given with the job number, the n oldest (from 1 to 32) action requests are printed for that job.

NOTE

Entering an account that does not have SWCFG privilege in the OPSER operator table with the OPERATOR command allows anyone logged in to that account access to all accounts. You must use caution when deciding which accounts you place in the OPSER operator table.

A.2.2 Message Types

OPSER displays two types of information on the Operator Services Console (OSC): messages and action requests. Messages are simply informative. For example, a message might tell you that a control file has been queued for batch processing. Action requests require an operator to perform some action before the job continues (for example, mounting a tape).

The MESSAGE command lets you define the level of information displayed at the OSC. For example:

- MESSAGE ALL—You want both messages and action requests displayed at the OSC.
- MESSAGE REQUESTS—You want only action requests displayed at the OSC. Messages are held in a table until you specify MESSAGE ALL.
- MESSAGE NONE—You do not want any messages displayed at the OSC. Messages and requests are held in a table until you specify MESSAGE ALL or MESSAGE REQUESTS and then the RETYPE command.

When OPSER starts, the default is MESSAGE ALL. If you suspend the display of messages, or messages and requests, by typing MESSAGE REQUESTS or MESSAGE NONE, the suspended items are held in a table. The table can hold a maximum of 32 messages or requests. (The actual number held depends on the length of the messages and requests.) If OPSER receives a message or request after the table is full, the new message or request is added to the table, and the oldest message or request is deleted.

Thus, you should not suspend message printing for long, or you may lose messages and requests. When you want to see messages and requests again, type MESSAGE ALL. Then use RETYPE to print the messages and requests currently held in the table.

If you see "Message Table Full" displayed at the OSC, it means that a message was received, and the table was full. Some unknown number of old messages and requests had been lost while MESSAGE NONE or MESSAGE REQUESTS were in effect.

To help you distinguish between messages and action requests on the OSC, output appears in distinct formats. A single TAB character denotes message text; two TAB characters denote action request text.

Message requests have the format:

```
MESSAGE         nnnnn : date time JOB:NN KB:MM jobname [p,pn]
        text
```

If the first character of the message text is a question mark (?) character, the words FATAL MESSAGE replace MESSAGE in the identification line.

Action requests have the format:

```
REQUEST         nnnnn : date time JOB:NN jobname [p,pn] rcvrid
                text
```

In an action request, the first character of text is a Ctrl/G (BEL) that sounds a bell in the terminal to alert the operator. Table A–2 summarizes the contents of messages and action requests.

Table A–2: OPSER Message and Action Request Contents

Item	Meaning
nnnnn	Sequence number of message starting at 1 and incremented by 1 during processing; use with the ANSWER command to respond to a specific action request. After restarting with currently existing work files, OPSER rounds the last assigned sequence number up to the next multiple of 10. This number is the next assigned sequence number.
date	Current system date.
time	Current system time of day.
JOB:nn	Job number under which sending program is running; use it, preceded by the number (#) sign, to refer to the job in an INTERRUPT, DELETE, or RETYPE command.
KB:mm	The keyboard number of the job that sent the message. The KB:mm field is displayed as DET if the job was detached, and as KB:?? if OPSER is unable to determine the keyboard number. Typically, the KB:?? message appears when the job has terminated.
jobname	Name of the program that an online sending job is running; this name is printed for information purposes. OPSER prints six question marks, ??????, in place of the jobname if it is unable to determine the jobname. Typically, the ?????? message appears when the job has terminated.
[p,pn]	Project-programmer number (PPN) under which the sending online job is running.
rcvrid	For action requests only. The message receiver identification given to the job when it is started. The identification can be standard (LPnSPL for the SPOOL program or BAnSPL for the BATCH program) or can be optional (as specified in the NAME startup switch for SPOOL or BATCH). Use the receiver identification with the INTERRUPT command to refer to the spooling program.

An action request is the result of a spooling job's requesting interaction with the operator: the spooling job sends a message to OPSER. The spooling job performs no further processing until operator action is taken or until the condition that generated the action request is satisfied. OPSER broadcasts action requests on the OSC unless the message level is set to NONE.

To respond to an action request, the operator uses the ANSWER command. Because OPSER usually runs detached, an operator can run PLEASE at valid operator terminals to send the response to OPSER. The ANSWER command requires a message number to identify the action request to which the operator is responding.

OPSER sends the text in the ANSWER command to the spooling program and deletes the action request. If the response satisfies the request, the program resumes processing. If the response does not satisfy the request, the spooling program still expects an appropriate response. The operator must then send such a response by means of the INTERRUPT command (see the section "Operator INTERRUPT Command").

A.2.3 Valid Operator and Online Job Lists

The LIST command displays data concerning the valid operator list or the online job table maintained by OPSER:

- LIST OPERATORS displays the keyboard number and PPN combinations currently defined as valid operators. The OPERATOR command updates the valid operator list. Initially, only users with SWCFG privilege can update the valid operator list.
- LIST JOBS prints the header ONLINE JOB and then prints data concerning online jobs in the following format:

  #n [p,pn] 'rcvrid' SL=n

Table A–3 describes the items in the online job list.

Table A–3: OPSER Online Job List

Item	Meaning
#n	The job number under which the online program is running.
[p,pn]	The PPN under which the online job is running.
'rcvrid'	The message receiver logical identification used to identify the online program in INTERRUPT commands.
SL=n	The OPSER shutdown level used when SHUTUP notifies OPSER to terminate spooling operations in an orderly fashion.

The job data listed is important to the operator. For example, certain OPSER commands require either a receiver identification or job number to communicate with a spooling program.

A.2.4 Operator INTERRUPT Command

The INTERRUPT command sends unsolicited messages and special commands to spooling programs. OPSER accepts any text in the INTERRUPT command and sends it to the destination job. Only the destination job interprets the text. If a response to an INTERRUPT command is generated, it is displayed on the OSC. Tables A-6, A-12, and A-15 summarize special text interpreted by spooling programs as commands.

The INTERRUPT command allows an operator to control and to monitor spooling programs. Each spooling program recognizes commands that can be divided into three sets:

- Commands that have no direct effect on how the program itself processes a request queued by a user. Included in this set of commands are PAUSE, CONTINUE, NOTICE, LAST, and STATUS. They allow the operator to control the program in general and to gain information about the program.
- Commands that do have a direct effect on program operation but are uniform for all spooling programs. END, ABORT, and OFFLINE represent this type of command which directly influences the job.
- Commands that are unique to the individual spooling program. A command for forms control on a line printer is an example.

A.2.5 OPSER Startup Procedure

Use commands in the system startup control file to start the OPSER program. See "System Startup Command File: START.COM" in Chapter 3 for information about creating a startup control file. This section describes the commands you should include in that startup file.

The following is a typical sequence of startup file commands to start OPSER:

```
$ RUN OPSER$:OPSER
LOG OPSER$:OPSER.LOG;ALL
CHA KB0:
MESS ALL
OPER KB*:[1,2]
DETACH
```

The LOGFILE, CHANGECONSOLE, MESSAGE, OPERATOR, and DETACH commands establish initial operating conditions for OPSER. The LOGFILE and MESSAGE commands set conditions normally established as defaults and are shown for clarity. An explanation of these commands follows:

- The LOGFILE command retains the file OPSER$:OPSER.LOG with a message level of ALL. As a result, all messages, action requests, OPSER information lines, and operator responses are written to the file to provide a complete historical reference.
- The terminal on which OPSER starts is automatically defined as the OSC. You can change this definition by placing the CHANGECONSOLE command in the startup procedure. Any valid operator may also alter the assignment o the OSC. It is most convenient to keep keyboard unit 0 as the OSC because, regardless of the number of logins currently allowed on the system, the operator can always use the system console terminal to communicate with OPSER.
- The MESSAGE command specifies that all messages, action requests, and OPSER information lines are printed at the OSC. Normally, the operator is concerned only with action requests. If the operator is concerned with any unanswered action request, the RETYPE command can be issued. In addition, the log file can be closed with the LOG command and printed at any time to recover recorded information.
- The OPERATOR command updates the valid operator list with user [1,2] on any keyboard. Allowing the operator to communicate from any terminal on the system gives flexibility to those who are performing operator functions.
- The DETACH command causes OPSER to detach itself from the terminal on which it is running. At the start of timesharing, this terminal is usually the system console terminal (KB0:). While OPSER runs detached, it is immune from tampering by unauthorized users. The operator can communicate with OPSER through the PLEASE program.

A.2.6 OPSER Action Under Various Startup Conditions

OPSER startup actions depend on the way OPSER last terminated:

- When OPSER runs after you have shut down operator services in an orderly fashion through SHUTUP, a normal startup procedure occurs. When shutting down in an orderly fashion, OPSER removes all entries from its online job table and closes all files properly. On restarting, OPSER finds all data valid. The online job table is empty, and entries in the valid operator table are legal and therefore retained.

- When OPSER starts after a system crash, it may find entries in its online job table. Because OPSER starts before other online jobs, none of the jobs in the table is active; therefore, the program clears the entries from the table. If any entries in the valid operator table are illegal, OPSER clears that table and sets one valid operator entry: the current user from any keyboard.
- When OPSER starts after an unexpected termination, it may find entries in its online job table for jobs still active on the system. In this instance, the program sends a command to each active job having an entry in the online job table. The command requests the job to retransmit the last message it sent to OPSER. OPSER does not wait for a response. If any job in the online job table does not meet OPSER's integrity checks, OPSER removes its entry from the table (takes the job off line) and generates an internal LIST JOBS command. OPSER also scans the entries in the valid operator table. If any entry is illegal, the program removes it. If OPSER removes any entry, it generates an internal LIST OPERATORS command.

A.3 Queue Manager Program: QUEMAN

The queue manager program QUEMAN maintains the file of queued requests (QUEUE.SYS in the OPSER$: account and communicates with spooling programs to execute queued requests. Queue management involves two modules stored in the OPSER Package Library: QUEMAN and QUMRUN. Both modules have a protection code of <232>, and require SWCFG privilege to run.

QUEMAN:

- Sets initial conditions on starting
- Checks initial conditions after restarting
- Chains to QUMRUN

QUMRUN:

- Executes commands
- Manages the queues

For simplicity, the documentation refers only to one program, QUEMAN, which functionally includes the two modules.

The QUEMAN program runs only if the job has SWCFG privilege and the OPSER program is running (that is, only if the name OPSER is in the system table of message receivers). Starting QUEMAN is a prerequisite to starting the spooling programs. To start QUEMAN, type the following command while logged into the system under an account SWCFG privilege:

```
$ RUN OPSER$:QUEMAN
QUEMAN V9  RSTS V9  TIMESHARING
STARTED AT: 01:52 PM ON 03-MAR-85
```

If you do not have SWCFG privilege, the system displays an error message to indicate that you do not have access to QUEMAN.

If the job has enough privilege, the program prints its identification line and startup lines and begins preliminary error checking.

QUEMAN tests for the presence of OPSER which must be started before QUEMAN can run. If OPSER is not running, QUEMAN prints an informational message and terminates:

```
QUEMAN cannot run without 'OPSER' active
```

QUEMAN opens the files QUEUE.WRK, OPSER1.WRK, and QUEUE.SYS in the OPSER$: account. If QUEUE.WRK does not exist, the program prints the following informational message and creates the file:

```
'QUEUE.WRK' not found - will initialize
```

If OPSER1.WRK does not exist, the program prints the following informational message and terminates:

```
OPSER1.WRK not found --- can't run
```

To recover, the operator must start the OPSER program before starting QUEMAN. If QUEUE.SYS does not exist, the program creates and initializes it. QUEMAN signals this action by generating the informational message:

```
No queue file found -- will initialize
```

If QUEUE.SYS exists, QUEMAN makes sure that it has write access to the file. If another program has write access to the QUEUE.SYS file, QUEMAN generates the following message:

```
Queue file opened by another program
Try again (Y/N) <N>?
```

The operator must determine which job has the file QUEUE.SYS open and must terminate that job. If you type Y in response to the question, QUEMAN retries the open operation. Typing N terminates QUEMAN.

When the OPEN operation successfully gains write access to QUEUE.SYS, QUEMAN declares itself a message receiver on the system. If QUEMAN is already declared by a different job, the program prints the message:

```
QUEMAN cannot declare itself as a receiver ... can't run
```

The operator must terminate the other job so it is removed from the message receiver table.

To make sure that the previous QUEMAN job closed the currently existing QUEUE.SYS file properly, the program checks a flag value in the file. If the flag is not properly set, the program prints the following informational message:

```
Queue file not closed properly --  now checking data for consistency
```

A later informational message signals that all integrity checks were successful:

```
Queue file data checked for consistency
```

If the flag is properly set, QUEMAN performs consistency checking but does not print any messages unless it finds problems. If any problems are found, QUEMAN generates the following informational messages:

```
Queue file data inconsistent - will initialize
Initialized
```

The INITIALIZED message indicates that QUEMAN has set all entries in the QUEUE.SYS file to their initial conditions and has removed all queued requests from the queues. See the section "QUEMAN Consistency Checking" for a description of the queue file consistency checks QUEMAN performs.

After completing all initial checks, QUEMAN prints the number sign (#) prompt indicating its readiness to accept a startup command or switch.

A.3.1 QUEMAN Startup Commands and Switches

Table A–4 lists the startup commands that QUEMAN recognizes.

Table A–4: QUEMAN Startup Commands

Command	Syntax and Meaning
DETACH	DET[ACH] Detaches QUEMAN from the terminal. You should run QUEMAN detached.
INITIALIZE	INI[TIALIZE] Sets all the entries in the QUEUE.SYS file to their initial conditions and sets initial conditions for the spooling queues. Any currently queued requests are lost.

Before QUEMAN executes the INITIALIZE command, it checks its online spooler table. If the table has an entry for a spooling job, QUEMAN prints the following informational message:

```
Spoolers on line -- can't initialize
#
```

QUEMAN also recognizes startup switches. One or more switches may appear following a command or may appear alone on a line in response to the prompt.

Table A–5 lists the startup switches that QUEMAN allows.

Table A–5: QUEMAN Startup Switches

Option	Syntax and Meaning
PRIORITY	/PRI[ORITY]:nnn Sets the job priority to nnn, where nnn can be from -120 to +120. Without the switch, the program automatically sets the priority to 0. You must have TUNE privilege to specify this qualifier.
RUNBURST	/RUN[BURST]:nnn Sets the job run burst to nnn, where nnn can be from 1 to 127. Without the switch, the run burst value currently assigned is used. You must have TUNE privilege to specify this qualifier.

If any data entered in response to the number sign prompt are not valid commands or options, QUEMAN prints the following informational message:

```
Invalid response -- text
```

The text is the data entered. The program prints the # prompt again.

To detach QUEMAN, the operator types the DET command in response to the prompt. For example:

```
#DET
Detaching...
```

The program then prints the DETACHING message and detaches itself from the keyboard. Note that you should always run QUEMAN detached.

While detached, the job receives commands from the operator through OPSER (see the following section). At no time should the operator enter Ctrl/C to the QUEMAN program; this may corrupt the QUEUE.SYS file.

A.3.2 QUEMAN Interrupt Commands

QUEMAN recognizes interrupt commands that allow an operator, through the INTERRUPT command of OPSER, to get information on queue manager processing and to control program operation.

Table A–6 summarizes these interrupt commands.

Table A–6: QUEMAN Interrupt Commands

Command	Option and Meaning
DIS[ABLE]	QUE[UING]<text> SPO[OLING]<text> ALL<text> Disables QUEUING or SPOOLING or both operations with the ALL switch. When you specify DISABLE QUEUING, the program QUE stops sending messages to QUEMAN and stops listing the queue. Any jobs already sent to QUEMAN are processed. If you specify DISABLE SPOOLING, QUEMAN prevents any further jobs from being sent to any spooler. Jobs sent to a spooler prior to the DISABLE command are processed to completion. Specifying DIS[ABLE] ALL tells QUEMAN to terminate both QUEUING and SPOOLING. You can include a message to the user while specifying the DISABLE command. Type the command, the switch, and then a space followed by the message. After QUEMAN processes the command, it prints the <text> message in one of the following formats: • Further QUEUING DISABLED <text> • Further SPOOLING DISABLED <text> • Further QUEUING AND SPOOLING DISABLED <text> If you ignore the <text> field, QUEMAN prints the default text, "by operator." The message appears when you try to queue a file.
ENA[BLE]	QUE[UING]<text> SPO[OLING]<text> ALL<text> Enables QUEUING or SPOOLING or both operations with the ALL option. When you specify ENABLE QUEUING, the program QUEUE begins to send messages to QUEMAN and starts listing the queue. If you specify ENABLE SPOOLING, QUEMAN begins to send jobs to the spoolers. Specifying ENABLE ALL tells QUEMAN to start both the QUEUING and SPOOLING functions. Use the <text> field to send messages to the user. Type the command, the switch, and then a space followed by the text of the message. After QUEMAN processes the command, it prints the message in one of the following formats: • QUEUING ENABLED - <text> • SPOOLING ENABLED - <text> • QUEUING AND SPOOLING ENABLED - <text> The default <text> message is "by operator." The message appears when you try to queue a file.

(continued on next page)

Table A–6 (Cont.): QUEMAN Interrupt Commands

Command	Option and Meaning
END	Terminates QUEMAN in an orderly manner and disables queuing and spooling. If any spooling programs are on line to QUEMAN, END sends to OPSER the informational message "Spoolers still online - will clear table" and clears the online spooler table. END then sends a command to OPSER that takes the QUEMAN entry out of OPSER tables. QUEMAN closes its work files and removes its job from the system.
LAS[T]	Sends to OPSER the most recent message generated by QUEMAN.
NEX[T]	quenam=reqnam Places the pending user request identified by reqnam at the head of the queue specified by quenam. A request name can contain a job name, PPN, and sequence number. Request names are found in the listing generated by the QUE program command L dev:, where dev: is the appropriate queue name. For example, LP:, BA:, or RJ: are valid queue names.
OFF[LINE]	Terminates QUEMAN as the END command does but does not generate a notification message.
STA[TUS]	Prints a brief report of spooling jobs on-line to QUEMAN.

When a command is received from the operator through OPSER, QUEMAN performs the action requested and either generates a message or an information line. OPSER formats the message or information line and displays it on the OSC. An example of this interaction between an operator, QUEMAN and the OPSER program is shown using the STATUS command. The following example assumes the operator is running the PLEASE program and the OSC is the keyboard on which PLEASE is running:

```
#/INT #6:STATUS
COMMAND SENT TO 'OPSER'
#
MESSAGE         29 : 10-MAR-85 10:48 AM  JOB:6 DET QUMRUN[1,2]
        1 SPOOLER(S) ON LINE -
(25) LP1SPL LP1: FORMS=NORMAL;
```

PLEASE passes the full command line to OPSER and notifies the operator before it reprints the prompt. OPSER recognizes the text INT as a request to send unsolicited text to a spooling job. The characters "#6" in the command line are recognized as the number of the destination job to which the text must be sent. QUEMAN receives the text STATUS, generates a response, and sends the response back to OPSER. OPSER formats a message and displays it on the OSC for the operator.

The STATUS command itself is the operator's way of getting information about jobs that are on line to QUEMAN. QUEMAN'S status printout gives the number of spooling jobs currently on line and supplies data on each job (job number within parentheses, receiver identification, physical unit being spooled, and program default conditions). If the particular spooling job is processing a request, the data in the printout includes the following information about the request: its name, the PPN under which it is queued, and its sequence number. The STATUS command also tells you if SPOOLING or QUEUING is disabled.

The NEXT command allows the operator to place a job request at the head of a queue. Job requests are identified by the name, account, and sequence number under which the request is queued. A request moved to the head of a queue is the next one QUEMAN sends to a spooling program servicing that queue. See the *RSTS/E Utilities Reference Manual* for a description of the NEXT format in the QUE program discussion.

If QUEMAN detects an error in the NEXT command, it generates an error message in the format:

```
'NEXT' CMD: string - text
```

The string is the command you entered. Table A–7 lists the possible error texts.

Table A–7: NEXT Command Error Text

Text	Meaning
ILLEGAL PARAMETERS	Only a queue name and request name, separated by an equal (=) sign, are allowed in the NEXT command.
ILLEGAL SYNTAX	A space or an = character is missing.
JOB IN PROCESS	The specified job request has already been sent to the spooling program.
MULTIPLE JOB SPECIFIED	The specified request name matches two or more jobs in the queue.
NO MATCH FOR JOB	The request name does not match any job in queue.

A.3.3 QUEMAN Startup Procedure

You should place commands in the system startup command file to start the QUEMAN program. See "Creating and Using Command Files" in Chapter 3 for information about creating a startup command file.

The following are typical commands to start QUEMAN:

```
$ RUN OPSER$:QUEMAN
/PRIORITY:0/RUNBURST=6
DETACH
```

The priority is set to zero because most jobs on RSTS/E run at priority minus eight and QUEMAN, running at priority zero, can more readily process the queue file. Include the DET command to make QUEMAN run detached on your RSTS/E system.

NOTE

> Without the PRIORITY switch, QUEMAN automatically sets its priority to zero. You can explicitly specify the switch to provide a record of what the program does automatically.

Because QUEMAN examines the QUEUE.SYS file when it starts, there is no need explicitly to initialize the file. If the file contains bad data, QUEMAN resets all entries to initial states. Therefore, you should specify the INI command only in situations in which you want to explicitly initialize the queue file.

A.3.4 QUEMAN Action Under Various Startup Conditions

QUEMAN startup actions depend on the way QUEMAN last terminated:

- When QUEMAN runs after you shut down operator services in an orderly fashion through SHUTUP, a normal startup procedure occurs. QUEMAN finds that the QUEUE.SYS file was closed properly. The program performs consistency checking but does not print any messages unless it finds problems. If the data is consistent, QUEMAN retains all job requests in the queue file and examines each request for completeness and status. Because QUEMAN normally starts before any spooling programs, there are no entries in the online spooler table. QUEMAN checks to see that this table is clear of entries.
- When QUEMAN starts either after the system crashes or after QUEMAN itself terminates unexpectedly, it finds the queue file improperly closed and generates a message to that effect. The program performs consistency checking on the queue file structure and on the data in the QUEUE.SYS file itself, generates messages telling the results, and reestablishes communication with all spooling jobs.

A.3.5 QUEMAN Consistency Checking

The QUEUE.SYS file has room for 250 queued requests. QUEMAN creates an entry for a queued request from an entry in a free list. A request for a line printer or batch queue requires one entry from the free list. A request queued with an AFTER date and time, however, requires two entries from the free list—one for the proper queue and one for the AFTER queue.

In performing consistency checking on the QUEUE.SYS file, QUEMAN initializes all entries if it finds one of the following conditions:

- An entry in the free list is also in a queue or AFTER list
- The root of the free list is outside the legal range
- A request queued with an AFTER date and time that has not expired does not have an entry in the AFTER queue

For other inconsistencies, QUEMAN either may remove a single entry or perform some related action. If an entry in the AFTER queue has no corresponding entry in a spooling queue, QUEMAN removes the AFTER entry. If an entry in either an AFTER or spooling queue is not completely set up, the program removes the entry. (QUEMAN may have terminated while it was processing the request.) To signal this event, QUEMAN generates a message in the format:

```
%LPn: 'reqnam' [p,pn];QUEUED JOB INCOMPLETE - REMOVED FROM QUEUE
```

If an entry to be killed still remains, QUEMAN removes it and generates a message in the format:

```
%LPn: 'reqnam' [p,pn];QUEUED JOB IN 'KILL' STATUS - REMOVED FROM QUEUE
```

If QUEMAN finds that an entry has been processed (sent to a spooling program) but not yet completed, it places the request in a hold status and generates text in the format:

```
%LPn: 'reqnam' [p,pn];JOB PREVIOUSLY SENT TO SPOOLER;
WILL BE PUT INTO HOLD STATUS
```

The requester can remove the job request from hold status.

QUEMAN checks the online spooler table in the QUEUE.WRK file for consistency. If the program finds the table empty, it generates the informational message:

```
# SPOOLERS ON LINE = 0; WILL CLEAR TABLE
```

If the program finds that the count of online spooling jobs is larger than the limit (16), it clears the table and generates the informational message:

```
ONLINE SPOOLER TABLE CORRUPT - WILL CLEAR TABLE
```

If QUEMAN clears the online spooler table but spooling jobs are still running, the operator must terminate those jobs through OPSER and restart each spooling program. This action allows QUEMAN to put each spooling job on line again properly.

For each spooling program found in the online job table, QUEMAN removes the entry and generates the informational message:

```
'rcvrid' (nn) FOUND ON LINE TAKEN OFF LINE
```

QUEMAN sends the job a message requesting that it declare itself on line to QUEMAN again. The spooling job does not respond to the message while it is processing a queued request. The answering message indicating that the job is again on line to QUEMAN does not appear until the job completes the current request. When QUEMAN receives the online declaration from a spooling job, it generates the informational message:

```
'rcvrid' (nn) PUT ONLINE
```

A.4 Line Printer Spooling Program: SPOOL

The line printer spooling program SPOOL runs without operator intervention and executes queued requests to transfer disk files to a line printer or terminal. The program consists of three modules:

- SPOOL—Establishes initial conditions on startup and checks initial conditions upon restart
- SPLIDL—Executes when no job is being printed
- SPLRUN—Prints any spooled file

Each module is stored in the OPSER package account OPSER$:, has a protection code of <232>, and requires SWCFG privilege to run. For simplicity, this appendix refers only to one program, SPOOL, which functionally includes the three modules.

The SPOOL program runs only if the user has SWCFG privilege and the OPSER and QUEMAN programs are running. To start SPOOL, type the following command while logged in to the system under an account with SWCFG privilege:

```
$ RUN OPSER$:SPOOL
SPOOL V9  RSTS V9  TIMESHARING
#
```

Typically, this is done in the system startup file.

At this point, the SPOOL program performs a number of preliminary checks before it prints the number sign (#) prompt. The remaining portion of this section describes these initial program checks. Once you are familiar with this procedure, you can then learn the options SPOOL accepts in response at its program prompt. A description of these SPOOL program checks follows.

If you do not have SWCFG privilege, the system displays an error message to indicate that you do not have access to SPOOL.

If the job has SWCFG privilege, SPOOL runs and prints its identification line and the # prompt. In response to the prompt, give a specification in the following format:

```
# logical device:/startup switch(es)
```

The logical device is the name of the queue and the unit number within that queue from which this copy of SPOOL takes job requests to execute. This name ordinarily corresponds to the physical device on which SPOOL prints the requests; the physical device, however, may be changed by the PHYSICAL startup option (see the following section "SPOOL Startup Options").

The logical device name must have the form LPn: or LP:, where n is a unit number from 0 to 7. If the name is LPn:, the spooling job prints only requests queued either to that explicit unit or to the general spooling queue LP:. If you specify LP:, this copy of SPOOL prints requests in the printer queue regardless of the unit to which they were originally queued. The logical device specified as the queue name must be a line printer; the device name given must not have a logical assignment to some other device.

NOTE

> The general spooling queue LP: is useful only on systems having line printers with similar characteristics.

You can use any combination of the startup switches described in the following section to condition the operation of the SPOOL program. A switch is formed by a slash (/) character and an option that may take an operand or other switches.

A.4.1 SPOOL Startup Options

Startup options alter default processing conditions. You use these options to:

- Spool output to a different physical device
- Change form information
- Control job environment

Table A–8 summarizes these options. Note that the symbol (_) is an underline character and not a space.

Table A–8: SPOOL Startup Options

Option	Syntax and Meaning
ASSIGN	/ASS[IGN] Reserves the physical device for this job. Without this option, SPOOL retries continually whenever it cannot gain access to the unit and is ready to print a job.
FORM	/FOR[M]:name switch Defines the current form according to a name (NORMAL is the default name) and switches as follows: • /ALI[GN] Causes SPOOL to execute a forms alignment procedure before processing any further queued requests. • /DFL[ENGTH]:nnn Declares the device form length as nnn lines, where nnn is a number from 1 to 127. The length is that understood by the hardware of the output device. For example, on a line printer in the United States, the standard length is usually 66 lines per page. • /HEA[DINGS]:n Causes SPOOL to print n (from 0 to 3) heading burst pages preceding each job request. Unless /NH accompanies a file specification in the QUE command, SPOOL also prints n burst pages preceding each file in a job request. The default value is 1. • /LEN[GTH]:n Defines the length of software form as n lines per page, where n is a number from 1 to 127. If this switch is not given, SPOOL assumes 66 lines per page. • /PAG[E_EJECT]:YES Indicates that the device to which this copy of SPOOL directs output has a hardware top of form capability and that the device interprets the formfeed character (ASCII 12) as a top-of-form command. This condition is the default case for line printer devices. • /PAG[E_EJECT]:NO Conditions the software to translate a formfeed character (ASCII 12) into the proper number of linefeed characters because the hardware does not recognize the character as a top-of-form command. This condition is the default case for SPOOL output directed to terminal devices by the PHYSICAL option. • /WID[TH]:n Defines the width of the heading burst page as n characters per line. The default value is 132.
LPFORM	/LPF[ORM]:YES Indicates that LPFORM characters are interpreted by the system software for the output device. During normal transfer, the program passes LPFORM characters unmodified to the device. This condition is the default for line printer devices.

(continued on next page)

Table A–8 (Cont.): SPOOL Startup Options

Option	Syntax and Meaning
	/LPF[ORM]:NO Indicates that the system software does not interpret LPFORM characters. During normal transfer, the program converts each LPFORM character to an appropriate number of line feed characters. This condition is the default for terminal devices. Note that this option affects only jobs queued with the /LPFORM switch. The SPOOL program does not automatically invoke /LPFORM processing.
NAME	/NAM[E]:rcvrid Places the specified identification in the system message receiver table and in OPSER's online job table. This identification rather than the default identification is used by the operator in OPSER commands to access this spooling job. This identification must not be in use by some other job.
PHYSICAL	/PHY[SICAL]:dev: Uses this device as the physical device for this spooling program. Requests in the queue specified by the logical device are printed on this device. The default value is the logical device specified as the queue name.
PRIORITY	/PRI[ORITY]:nnn Sets the job priority to nnn, where nnn is a number from -120 to 120. Without this option, the priority remains as set by LOGIN. You must have TUNE privilege to specify this qualifier. Do not use /PRIORITY after a /FORM, /NAME, or /PHYSICAL option, or after an LPn: designator.
RUNBURST	/RUN[BURST]:nnn Sets the job run burst to nnn, where nnn is a number from 1 to 127. Without this option, the run burst remains as set by LOGIN. You must have TUNE privilege to specify this qualifier.

The PHYSICAL option alters the physical device on which queued requests are printed. For example, requests queued for line printer unit 1 could be processed on another line printer unit or on a keyboard unit. In any event, the actual device differs from the one for which the requests were queued.

The ASSIGN option reserves the actual spooled device to the job. Without this option, SPOOL periodically tries to gain access to the spooled device whenever SPOOL is ready to print and another job has ownership of the device. When SPOOL terminates, it deassigns the device.

The NAME option allows the receiver identification to be different from that normally assigned by SPOOL. The identification should be from one- to six-alphanumeric characters and must not exist in the system message receiver table. The identification must not begin with a number because the operator uses either this identification or a job number to identify a spooling job.

The FORM option allows the default form information to be altered. For example, individual characteristics of the NORMAL form can be altered, a new form with a different name can be defined, or a forms alignment can be requested. A forms alignment can be requested by itself or in conjunction with a form alteration. When a form name other than NORMAL is in effect, user's queuing requests must explicitly specify the form name to have files printed with its characteristics.

To alter the default characteristics of the NORMAL form, the operator can specify the appropriate auxiliary switches without the /FORM: switch. For example, to change the number and width of heading burst pages for line printer unit 0, type:

```
LP0:/HEADINGS:2/WIDTH:80
```

When executed, this command causes SPOOL to output two heading burst pages with a width of 80 characters per line print for each job request and each file within a request. The width used applies only to the burst pages, not to the data being printed.

Specifying a form with a name other than NORMAL usually means that the operator must load special paper in the output device. The /ALIGN switch with the FORM option requests a forms alignment. The operator then must align the form at system startup time. A SPOOL interrupt command with the /ALIGN switch can request a form alignment during timesharing. The section "Changing and Aligning Forms" describes the forms alignment procedure.

The following message can occur when the line printer queues are empty, but jobs with nondefault form names exist:

```
NO JOBS WAITING WITH FORMNAME 'XXXXXX' FOR SPOOLER 'LPn:'
OTHER JOB(S) WAITING.
***PLEASE INSPECT QUEUE AND TAKE APPROPRIATE ACTION***
```

The operator must then decide whether or not to use an interruption command to change the form name so the other jobs can be queued.

The PAGE_EJECT option determines whether the SPOOL program counts lines to effect forms control. (The DFLENGTH option can modify the effect of the PAGE_EJECT option.) A value of NO with the PAGE_EJECT option indicates that the output device (driver) does not understand a form feed character; that is, a form feed character does not cause the device to position itself at top of the (paper) form. With NO in effect, SPOOL counts lines to effect forms control. A response of YES with the PAGE_EJECT option indicates that the device (hardware) translates the form feed character to an appropriate number of line skips to place the device at top of form. With YES in effect, SPOOL need not count lines and may be able to pass all forms control data unaltered to the device driver. (The DFLENGTH option can modify the effect of YES.)

NOTE

SPOOL issues three forms-control characters:

- Form feed character (ASCII 12)
- Carriage return character (ASCII 13)
- Line feed character (ASCII 10)

No other forms-control characters are used. SPOOL has no provisions for changing or extending the set of control characters for any special device(s).

The terms line feed and line skip are not synonymous. Line feed refers only to the line feed character, which either SPOOL or the device driver sends to a device. Line skip refers to the movement of paper that the device itself effects.

If NO is in effect for the PAGE_EJECT option, SPOOL ignores any value specified in a DFLENGTH option. The operator should specify /PAGE_EJECT:YES for any terminal device that has a top-of-form capability.

SPOOL uses the value specified in the DFLENGTH option only if /PAGE_EJECT:YES is in effect. The /PAGE_EJECT:YES switch can be in effect either through default (on a line printer) or by explicit specification (for a terminal device having the form feed capability). The operator may specify the DFLENGTH option either at startup time in a SPOOL startup switch or during processing in a FORM interrupt command. For proper formatting, the DFLENGTH value must be the maximum number of lines that the output device skips when it receives a form feed command. The number of lines is called the device form length. This number may differ from the length of the paper form being used.

Different devices have different form lengths. Three examples can show how the device form lengths vary:

- The LA180 device has a switch labeled "Length of Form," which has settings for 4, 8.5, and 11 inches. These settings correspond to 24, 51, and 66 lines per form. (The number of lines per form given for each setting assumes the standard 6 lines per inch on the LA180.) If, when starting the spooling job, the operator sets the switch on the LA180 to 8.5 inches, the operator should also specify the DFLENGTH switch with a value of 51. If, at some later time, the operator changes the setting on the LA180 to 4 inches, the operator should also specify a new DFLENGTH option with a value of 24 in a FORM interrupt command to the spooling job.
- Many line printers have only one form length. If the hardware length is 66, the value in the DFLENGTH switch should be 66. If the hardware form length is 51, the operator should specify DFLENGTH=51 when starting the spooling job. (The default value for DFLENGTH is 66.)
- Some printing devices have an adjustable top-of-form. On these devices, the operator can adjust the device to execute an arbitrary number of line skips when it receives a form feed character. This adjustment is made most often on terminal-type devices with high quality print characteristics. When resetting the top-of-form for this type of device, the operator should also specify the DFLENGTH and LENGTH options with the new form length information. The operator should also request a forms alignment for each new form.

The effect of the DFLENGTH option on the SPOOL program differs for line printers and terminal devices:

- For a line printer, the DFLENGTH value affects only the procedure SPOOL invokes when it receives an ABORT command while it is processing a job request. If the DFLENGTH value equals the current length of paper form, the program issues a form feed character to position the device at the top of the next form. If the values are not equal, the program automatically invokes the forms alignment procedure to reestablish the top-of-form position.

  The automatic forms alignment procedure requires operator action to align the form and to respond to requests SPOOL makes. This is necessary only when the program has no way to determine where the top-of-form lies after an ABORT operation.
- For a terminal, the DFLENGTH value controls the way SPOOL keeps its place on a form and the way the operator handles recovery procedures. If the value of the PAGE_EJECT option is YES, the equality of the device form length and the paper form length is important. If the device form length and paper form length values are not equal, the SPOOL program counts lines to maintain a record of its position on a form and issues an appropriate number of line feed characters to simulate form feed. If these values are equal, SPOOL does not count lines and simply sends a form feed character to position the paper at top-of-form.

NOTE

> A line printer is a device whose interface to the computer is through a line printer controller and whose designation is in the form LPn:. A terminal is a device whose interface to the computer is through a keyboard line and whose designation is in the form KBn:. This distinction is critical because some devices such as the LA180 are called printers but may be connected to the computer through a keyboard line.

If the value in effect for DFLENGTH does not correctly match the setting of the hardware device, two undesirable conditions exist:

- SPOOL must incur the overhead of counting lines.
- Forms alignment may be lost during processing.

For these reasons, Digital highly recommends that the operator supply a new value of DFLENGTH when changing the hardware setting on the device.

A.4.2 Line Printer Spooling

If the physical device specified at startup time is a line printer unit, SPOOL applies the default values. Table A–9 lists these default values.

Table A–9: SPOOL Line Printer Default Values

Switch	Result
/DFLENGTH:66	The device form length is 66 unless otherwise specified. If you have patched the monitor to apply a different default form length, you must always specify the correct value in a DFLENGTH option.
/FORM:NORMAL /HEADINGS:1 /LENGTH:66 /WIDTH:132	The values shown are used as the form definition.
/LPFORM:YES	For any line printer, SPOOL assumes that the device driver properly processes the special LPFORM characters (see the RSTS/E Programming Manual).
/PAGE_EJECT:YES	For any line printer, SPOOL assumes that the device driver properly processes top of form characters.

You can also set the characteristics of a line printer with the SET PRINTER command (see Chapter 11).

A.4.3 Keyboard Spooling

If the physical device specified at startup is a keyboard device, SPOOL applies the default values. Table A–10 lists these default values.

Table A–10: SPOOL Keyboard Default Values

Switch	Result
/DFLENGTH:66	DFLENGTH has no effect unless PAGE_EJECT is explicitly specified as YES.
/FORM:NORMAL /HEADINGS:1 /LENGTH:66 /WIDTH:132	The values shown are used as the form definition.
/LPFORM:NO	The terminal driver does not process LPFORM characters. For any file that contains LPFORM characters, SPOOL simulates the LPFORM effect by issuing linefeed characters.
/PAGE__EJECT:NO	Most terminals do not have a hardware top-of-form capability. SPOOL simulates top-of-form by issuing the correct number of linefeed characters.

For terminals with the hardware top-of-form capability:

- Set the value of PAGE_EJECT to YES.
- Set the value for DFLENGTH to the actual device form length (see "SPOOL Startup Options").
- Set the proper characteristic of the device that accepts and processes a form feed. Use the SET TERMINAL command to perform this operation (see Chapter 8).

A.4.4 Startup Error Processing

If you make an error when responding to the number sign (#) prompt, SPOOL prints a message and the unparsed command line in the format:

```
?ERROR MESSAGE
unparsed command line
```

The unparsed command line is the right-hand part of the response beginning at the element causing the error. Table A–11 summarizes the possible messages.

Table A–11: SPOOL Syntax Error Messages

Message and Meaning	
?Can't parse remaining string	Something illegal or undefined was found in the command line. An undefined switch or missing slash (/) character causes this error.
?Duplicate switch	Two occurrences of the same switch were found in the command line.
?Illegal logical device	The logical device specified was not in the form LP: or LPn: (where n is a number from 0 to 7).
?Illegal operand	An illegal operand was found in an option or switch. For example, specifying nonnumeric characters in a switch that requires a number (/WIDTH:n) generates this error.
?Illegal physical device	The device specified in the PHYSICAL option is not a line printer or keyboard device.
?Missing operand	A switch requiring an operand was specified without one. For example, if a receiver identification is missing from the NAME option, this error is generated.

If SPOOL encounters no syntax errors in the response, it begins setting initial conditions. SPOOL reports errors in this phase by printing a message in the format:

```
ERROR IN SOME OPERATION  --  RESTARTING
error message
```

The program reprints its identification line and the prompt.

An error message ?Duplicate receiver ID means that the receiver identification, either the default one or the one specified in a NAME option, is already defined for another job. The operator should type the command line again and specify a unique receiver identification in the NAME option.

The error message ?No room in receiver table means that a general small buffer is not available to let SPOOL declare itself a receiving job. A later retry with the same command line should succeed.

The following message means that SPOOL attempted to send a message to that program and failed:

```
*****OPSER HUNG***** or *****QUEMAN HUNG*****
```

Either the named program does not have an entry in the system message receiver table or the program is not processing its messages and its message limit has been reached. The operator must determine the cause of the problem and restart the named program.

A.4.5 SPOOL Interrupt Commands

The operator communicates with a SPOOL job by means of INTERRUPT commands sent through the operator services program OPSER. Note that any responses to an INTERRUPT command are displayed on the Operator Services Console.

Table A–12 summarizes these SPOOL interrupt commands. Note that the circumflex (^) character marks the location of a required space.

Table A–12: SPOOL Interrupt Commands

Command Syntax	Meaning
ABO[RT]	Immediately terminates the current process and removes the request from the queue.
CON[TINUE]	Wakes up the spooling job to continue processing after a PAUSE command.
END	Closes out processing after completing the current request. The operator must then run the spooling program again to process further requests.
FOR[M]^name/switch	Changes the current output form to one identified by name and defined by switches. These switches change current form characteristics; those characteristics not changed by an option remain at their current definitions. See the startup option FORM description in Table A-8 for the allowable options. Alignment is done only if the /ALIGN switch is included.
FOR[M]	Displays characteristics of the current form if no name or options are in the command.
FOR[M]^/ALIGN	Requests a forms alignment procedure for the current form.
LAS[T]	Prints the most recent message generated by the spooling job.
OFF[LINE]	Immediately terminates all processing by this spooling job (same as ABORT followed by END).
PAU[SE]	Places the spooling job in a sleep state, during which it responds to most commands and resumes normal processing in response to a CONTINUE command.[1]
REQ[UE]	Stops processing the current request and replaces the request in the queue so that processing later resumes at the terminating point.
RES[TART]	Reprints the current copy of a file from the beginning, including heading burst pages.
RES[TART]^JOB	Reprints the current iteration of the job, including heading burst pages.
RES[TART]^:n	Reprints the current copy of the file starting at page n. If n is 0, restarts from the beginning of the file but omits the heading burst pages.
STA[TUS]	Prints a status report for the spooling job.

[1] Note that processing continues if any job modification command such as ABORT or REQUE is sent.

A.4.6 SPOOL Startup Examples

This section shows four examples of the startup procedure. They represent typical cases.

A.4.6.1 Line Printer Startup with All Defaults

In response to the SPOOL program prompt, the operator types:

```
#LP0:
```

If SPOOL finds no errors, it prints the message:

```
DETACHING...
```

SPOOL detaches itself and leaves the terminal logged off the system.

The following are the characteristics of this copy of SPOOL:

- The program prints only job requests that were queued to line printer unit 0 with a form name of NORMAL or queued to the general line printer queue with a form name of NORMAL.
- The program prints on line printer unit 0.
- The program prints one heading burst page before each job and, unless a file is queued with /NH, prints one heading burst page before each file. The widtl of the burst page is 132 columns, and the form length is 66 lines.
- The program assumes the device driver handles LPFORM characters.
- The program assumes the device handles top-of-form.
- The program assumes the device form length is 66 lines.
- The run burst and priority of SPOOL are those in effect when the job starts.
- When the program receives a job from QUEMAN, it attempts to assign the printer. If the assignment fails, the program retries periodically until it succeeds. When it finishes processing a job request, the program deassigns the device.

The operator may later change all form characteristics (name, length, width, device form length, and headings count) through appropriate interrupt commands.

A.4.6.2 Line Printer Startup with Narrow Width

In response to the prompt, the operator types the command line:

```
#LP1:/FORM:NARROW/WIDTH:80/RUNBURST:12/ASSIGN
```

The following are characteristics of this copy of SPOOL:

- The program prints only jobs queued with a form name of NARROW and queued to either unit 1 or the general line printer queue.
- The program prints on the line printer 1 device.
- The program prints one heading page of width 80 columns; form length is 66.
- The program assumes the device handles top-of-form.
- The program assumes the device driver handles LPFORM characters.

- The program assumes the device form length is 66.
- The priority of the job remains unchanged. The program sets the run burst to 12.
- The program does not begin spooling until it can assign line printer unit 1 and does not deassign the device until it terminates.

In this example, assume that line printer unit 1 is a relatively slow device and that the SPOOL program can generate output data for this unit faster than the device can print. Optimal use of the device results if the unit keeps running at full speed for the entire job request it is printing.

To effect optimal use, it is sometimes necessary to raise either the priority or run burst of the SPOOL program. The priority of a job generally affects how often it runs, whereas the run burst affects the length of time the job runs once it starts running. Because the SPOOL program is usually I/O bound and waiting for the device to complete printing, there is little reason for it to run more often. Changing the priority is usually unnecessary and ineffective. Changing the run burst, however, may improve processing.

When SPOOL starts executing, it remains in the run state until one of the following events occurs:

- The job's run burst expires. That is, the job actually executes for its full run burst.
- The job requests some type of I/O and the system cannot immediately satisfy the request.

For the spooling program, the second event is much more likely because the system buffers only a fixed amount of data before it refuses to satisfy further requests for output. When the refusal occurs, the system activates another job and the spooler must wait until the system sends at least some of the already buffered data to the output device. (This wait condition shows as an LP or TT STATE on a SYSTAT listing.)

As long as the buffers for the device are never completely emptied, the system keeps the device running at or near full speed. To a certain point, raising the spooling job's run burst makes it more likely that the job will continue running until it completely fills the buffers for its output device. After a certain point, raising the run burst has little benefit because the job can run only until the buffers are full. Conversely, decreasing the run burst causes the system to deactivate the program before the buffers are full. It is more likely that the printer can empty the buffers before the system reactivates the job.

If a spooling job spends a significant time in the RN state while it is actually printing, the run burst is too low. DIGITAL recommends that you raise the run burst to a value at which the job spends most of its time in an output wait state.

A.4.6.3 Keyboard Startup on an LA36

For this example, assume that keyboard unit 5 is an LA36 terminal. At the prompt, the operator types a command line as follows:

```
#LP2:/PHYSICAL:KB5:/HEADINGS:0/ASSIGN
```

The following are the characteristics of this copy of SPOOL:

- The program prints only jobs queued with a form name of NORMAL and queued to the unit 2 or the general line printer queue.
- The program prints on keyboard unit 5.

- The program prints no heading burst pages. The form length is 66.
- The program assumes that keyboard unit 5 does not handle top-of-form; therefore, it counts lines as it prints them to simulate top-of-form.
- The program assumes the device driver does not handle LPFORM characters; therefore, it simulates LPFORM with line feeds.
- The run burst and priority remain unchanged.
- The program assigns keyboard unit 5 at startup and keeps it assigned until it terminates.

Because the LA36 is a slow device, processing is not slowed any further with burst pages. The /HEADINGS:0 switch suppresses printing of burst pages.

To retain any forms alignment on the LA36, the device is permanently assigned to the spooling job. This assignment prevents any other job from using the device and leaving the carriage at a position other than the top of form position. When the spooler starts processing a request, it must assume the device is at its top of form position because the program has no way to force the device to advance to some known position. If the form is not aligned properly when SPOOL starts the output, the entire output is misaligned.

A.4.6.4 Keyboard Startup on an LA180

For this example, keyboard unit 2 is a serial LA180 DECprinter. This terminal has a hardware top-of-form capability. In addition, the form length selector switch on the terminal is set to 8.5 inches (51 lines per form) and the paper form is also 51 lines per page.

In response to the prompt, the operator types a command line as follows:

```
# LP3:/PHYSICAL:KB2:/PAGE_EJECT:YES/DFLENGTH:51/LENGTH:51-
MORE>    /WIDTH:40/FORM:SPECIAL
```

The following are the characteristics of the SPOOL copy:

- The program prints only requests queued with a form name of SPECIAL and queued to either the unit 3 or the general line printer queue.
- The program prints on keyboard unit 2.
- The program prints one heading page that is 40 columns wide and 51 lines in length.
- The program assumes the device handles top of form.
- The program assumes that the device driver does not handle LPFORM characters. The program therefore simulates LPFORM with line feeds.
- The run burst and priority remain unchanged.
- The program assigns the keyboard before it prints each request and deassigns it after printing the request.

Because:

- The LA180 is a much faster device than the LA36, SPOOL can print a small number of burst pages to simplify distinguishing jobs and files. The narrowest width for the burst page is 40 columns.
- The paper form length and the device form length are equal, SPOOL can position the device to top-of-form by issuing a form feed character.
- The program can position to top-of-form, it does not need to keep the unit assigned.

The ASSIGN option, therefore, does not appear in the example. If the paper form length did not equal the hardware setting, however, SPOOL should keep the device assigned to preserve the top-of-form position.

A.4.7 Recovery from Line Printer Errors

The interaction between the spooling job and the OPSER program controls error handling in the SPOOL program. For example, if the paper runs out or jams, SPOOL receives the error message:

```
?Device hung or write locked
```

SPOOL discontinues processing and sends OPSER a message. OPSER generates a message for the operator:

```
MESSAGE        3 : 28-MAR-85 04:33 PM  JOB:21  DET SPLRUN[1,100]
            LP0   : HUNG  --    JOB: [220,40]USER1
(PUT DEVICE ONLINE TO CONTINUE)
```

The message tells the operator that the spooling job for line printer unit 0 has a problem and that the device needs to be made ready again before processing can continue.

For a SPOOL program servicing a line printer, there are certain errors from which you can easily recover. The device handler for the line printer tests the ready status of an offline unit every 10 seconds. If the operator corrects the error (for example, fixes a paper jam) and puts the line printer on line again, the software detects the ready status and continues printing buffered characters. The SPOOL program finds that the error condition is cleared and continues processing.

If a special form is being processed and SPOOL receives an error, SPOOL may request the operator to realign the forms before it continues processing on its own.

If the operator cannot correct the error or wants to perform some other operation, the spooled device may be left off line. This prevents the software from reprinting before the program processes an operator request. The operator uses the INTERRUPT command with the proper text to access the spooling job.

SPOOL allows the operator several ways to control the restart of a queued request. The RESTART command can restart:

- The current copy of a queued request (RESTART JOB)
- The current copy of a file in a queued request (RESTART)
- The current copy of a file at a specific page (RESTART:nnn)

The REQUE command can reenter the current job in the queue for later processing. On receiving a RESTART command, the program waits for any currently buffered data to be printed before actually restarting.

The REQUE command causes SPOOL to remember the point at which processing was interrupted. After resuming the requeued request, SPOOL scans for the page on which processing terminated and continues printing at the start of that page. An error may occur if a user modifies the requeued request before SPOOL can resume the printing.

If the operator responds to an error condition with an ABORT command, which terminates the job, SPOOL clears the output buffers on the line printer unit. If the device is a terminal, SPOOL prints any buffered data. The program then issues a message to show that it is terminating.

Before terminating a queued request, the SPOOL program makes sure that all buffers have been emptied successfully.

A.4.8 Line Printer Output

SPOOL generates job header and file header burst pages to identify print requests and files within a print request. Both types of header page contain identification and general accounting information as follows:

- The identification consists of large, easily readable block letters created from the character generation file CHARS.QUE. The job identification contains the account number of the user requesting the job and the name of the job. If no job name appeared in the QUE command, SPOOL prints the name of the first file in the request as the job name. The file identification shows the file name and type.
- General accounting information for the job header is on the burst page and is offset from the identification by two rows of special characters. The accounting information contains five lines of data:
  - The first line gives the job name, current date, current time of day, and the requester's account number.
  - The second line gives the date and time of day of the request and the device for which the request was queued.
  - The third line gives the system name.
  - The fourth line gives the QUE options used to process the request.
  - The fifth line gives the job copy number.

  The accounting information for the file header burst page appears below the identification and is framed (above and below) by two rows of special characters. The first line of information gives the job name used when SPOOL printed the file and shows the current date, time of day, and account as the job header burst page does. The second line gives the copy number, the QUE options used on the file, and the complete specification of the file. The third line gives the record type and carriage control format used to print the file.

If SPOOL does not print the file because of an error, file header identification and accounting information is replaced by an error message framed (above and below) by five rows of special characters. The error message is standard RSTS/E error text.

A.4.9 Error Messages During User Output

SPOOL reports errors it receives during printing in the line printer output. SPOOL differentiates the error message from requested output by framing the text (above and below) with five rows of special characters.

Table A–13 shows the possible SPOOL error messages and related meanings.

Table A–13: SPOOL User Requested Output Error Messages

Message and Meaning

?Can't find file or account

The related file was deleted between the time it was queued and the time SPOOL tried to print it.

?File restart requested on page n

The operator requested a restart at the indicated page or placed the job request back in the queue with the REQUE command. If the operator restarted the job request, SPOOL scans the file to find the correct page and continues printing on a new page following the message. If the operator requeued the job, this message appears when SPOOL restarts the job.

?Job aborted

Either the operator terminated the job request with an ABORT command or the K command of the QUE program was issued while the job was being printed.

?Job aborted because of file error(s)

Errors in some file in the job prevented SPOOL from printing the remainder of this job.

?Job requed at page n

The operator placed this job request back in the queue with the REQUE command. SPOOL later continues the job request with a new header burst page and the text FILE RESTART REQUESTED ON PAGE n, where n is the same page number shown in the REQUE message.

?<error text>

SPOOL encountered the RSTS/E error shown, aborted processing the request, and cleared the request from the queue.

A.4.10 Changing and Aligning Forms

The FORM command can:

- Inform the spooler of a change in form on its device
- Request an alignment operation on the device
- Display information about the current form

Switches with the command specify the new form definition and request alignment. The command may request alignment when a new form is specified.

To request a form change, the operator sends the FORM command with the form name and the switches to define the form to the related spooling job. If an alignment is wanted for the current form, the operator can include the /ALIGN switch with the FORM command. The SPOOL program does not recognize the form change request until it becomes idle. If it is processing a queued request, it completes the processing before it performs the forms change. (To make the change immediately, request that SPOOL requeue the current request. The program requeues the request and becomes idle.)

To perform only an alignment, specify the FORM command without any form name and with only the /ALIGN switch. Again, the command takes effect only when SPOOL becomes idle. The program follows these steps:

1. Prints characters to delineate left and right margins, and prints the text TOP OF FORM centered on the same line.
2. Generates characters to position the paper on the last line of the form.
3. Prints characters to delineate the left and right margins, and prints the text END OF FORM centered on the same line.
4. Positions the paper at top of the next form.

SPOOL generates an action request and awaits a response. The operator may inspect the alignment and type one of two responses to the action request. If the alignment is not correct, the operator can adjust the hardware and send the response RETRY back to the spooler. In response to RETRY, the spooler performs steps 1 through 4 again and generates another action request. When the alignment is accurate, the operator types GO as a response to the action request. The spooler returns to normal processing.

Whenever SPOOL processes any FORM command, it sends a message to the operator indicating that the new form characteristics in effect. In any FORM command, only those options specified in the command change the form characteristics. Thus, if the operator sends a FORM command alone, without a form name or any options, the spooler displays the characteristics of the current form on the Operator Services Console.

A.5 Batch Processor Program: BATCH

The BATCH system program runs without user intervention on pseudokeyboards and executes files of standardized commands queued for either a specific or the general Batch Processor. The program consists of four modules:

- BATCH sets initial conditions for batch processing.
- BATIDL checks status when no requests are being handled.
- BATDEC decodes and performs error checking on command files.
- BATRUN executes the decoded batch commands.

For simplicity, this appendix refers only to one program, BATCH, which functionally includes the four modules. Each module is stored in the OPSER package account OPSER$:, has a protection code of <232>, and requires SWCFG privilege to run.

To run BATCH, type the following command while logged in to the system under an account with SWCFG privilege:

```
$ RUN OPSER$:BATCH
BATCH V9  RSTS V9  TIMESHARING
#
```

Typically, this startup of BATCH is done in the system startup file.

If you do not have the SWCFG privilege, the system displays an error message indicating that you do not have access to BATCH.

If you have the SWCFG privilege, BATCH runs and prints its identification line and the number sign (#) prompt. In response to the # prompt, enter a specification in the following format:

```
logical device:/startup switch(es)
```

The logical device is the name of the queue and the unit number within that queue from which this copy of BATCH takes job requests. This name does not specify the static pseudo keyboard on which BATCH executes requests. The PHYSICAL startup option can specify the static pseudo keyboard unit on which BATCH executes. The logical name must have the form BA: or BAn: where n is a unit number from 0 to 7. If BAn: is specified, the Batch Processor executes requests queued to either that explicit unit or the general batch queue BA:. If BA: is specified, the program executes job requests in the batch queue regardless of the unit to which they are queued.

NOTE

> The general batch spooling queue, BA:, is useful only on systems with Batch Processors running with the same default conditions.

The logical device specified as the queue name must be a Batch Processor; the device name given must not have a logical assignment to some other device.

If only a queue name is given in response to the prompt and BATCH finds no errors, the program establishes default processing conditions, prints the DETACHING message, and detaches itself from the terminal.

BATCH forms the default receiver identification from the queue name. BATCH also establishes other default conditions:

- BATCH tolerates no errors and terminates any queued request encountering a fatal or warning error.
- User log files are queued to the system default line printer with deletion specified.
- The run burst remains unchanged.
- The priority remains unchanged.

At startup, BATCH performs the same error checking and processing SPOOL performs. See the section "Startup Error Processing" for the description of SPOOL startup error processing.

A.5.1 BATCH Startup Options

Include options with the BATCH processor designator to control its operation.

Table A–14 summarizes these options, which are in the form of switches.

Table A–14: BATCH Startup Options

Option	Syntax and Meaning
ASSIGN	/ASS[IGN] Reserves a pseudo keyboard device to this job.
ERROR	/ERR[OR]:severity Sets the error severity default for this processor so that errors of that severity are tolerated and queued requests having that or lesser severity end successfully.
	/ERR[OR]:FAT[AL] Tolerates all errors.
	/ERR[OR]:WAR[NING] Tolerates only warning errors.
	/ERR[O]R:NON[E] Tolerates no errors (fatal and warning errors terminate queued requests). This is the default severity.
DELETE	/DEL[ETE] Deletes user log files after printing. This is the default condition unless the NODELETE option is specified.
NAME	/NAM[E]:rcvrid Places, in the system message receiver table, the specified receiver identification for this job instead of the default identification. The default for a Batch Processor is BAnSPL for specific processors and BASPL for the general processor.
NODELETE	/NODEL[ETE] Does not delete user log files after printing. Without this option, all log files are deleted. Requester may override this option with the /NOQUE switch in the $JOB command.
NOQUEUE	/NOQUE[UE] Does not queue user log files for printing. Without this option, all log files are queued without a queue name and with deletion unless changed by the /QUEUE and /DELETE switches. Requesters cannot override this option.
PHYSICAL	/PHY[SICAL]:dev: Uses the pseudo keyboard (for example, PK1:) as the device on which this processor executes user batch jobs.
PRIORITY	/PRI[ORITY]:nnn Sets the job priority of BATCH to nnn, where nnn is a number from -120 to 120. Without this option, the priority remains unchanged. You need TUNE privilege to specify this qualifier. Do not use /PRIORITY after a /NAME, or /PHYSICAL switch, or after a BAn: designator.

(continued on next page)

Table A–14 (Cont.): BATCH Startup Options

Option	Syntax and Meaning
QUEUE	/QUE[UE]:[quenam] Queues all user log files to the specified spooling queue (LP0: through LP7:) rather than to the general print queue. If no queue name is specified, all jobs are queued to the general print queue. Files are deleted after printing. Requesters may override queuing with the /NOQUE switch in the $JOB command. Deletion of the files by the SPOOL program may be overridden by including /NODELETE as a startup switch.
RUNBURST	/RUN[BURST]:nnn Sets the job run burst of BATCH to nnn, where nnn is a number from 1 to 127. Without this option, the run burst remains unchanged. You need TUNE privilege to specify this qualifier.

A.5.2 BATCH Interrupt Commands

The operator uses interrupt commands sent through the operator services program OPSER to communicate with a Batch Processor. Note that any responses to an interrupt command are displayed on the Operator Services Console.

Table A–15 summarizes these BATCH interrupt commands.

Table A–15: BATCH Interrupt Commands

Command Syntax	Meaning
ABO[RT]	Immediately terminates the current process and removes the request from the queue.
CON[TINUE]	Wakes up the Batch Processor to continue normal processing after either a PAUSE command or after a user request for operator action (for example, in the $MESSAGE/WAIT command).
END	Terminates processing after completing the current request. The operator must then run the Batch Processor again to process further requests.
LAS[T]	Prints the most recent message generated by this Batch Processor.
NOT[ICE] text	Inserts the specified text in the user log file and precedes it with the heading: NOTICE FROM OPERATOR.
OFF[LINE]	Immediately terminates all processing by this Batch Processor (same as ABORT followed by END).
PAU[SE]	Places the Batch Processor in a sleep state, during which it responds to most commands and resumes normal processing in response to a CONTINUE command.
STA[TUS]	Prints a status report for this Batch Processor.

A.5.3 BATCH Startup Procedure

BATCH is most conveniently started by commands in the system startup command file. See the section "Creating and Using Command Files" in Chapter 3.

A.5.4 Operator Action Requests from BATCH

The BATCH program, in processing certain commands in a user command file, generates requests for operator action. These requests come through OPSER as action requests that require some operator action and a typed answer before the Batch Processor continues processing. These action requests result from commands requiring the operator either to mount and dismount volumes or to respond to a special message with which a stall is involved. The operator answers an action request with the ANSWER command through OPSER.

On a mount request, BATCH asks the operator to mount a device by an action request with text in the following format:

```
MOUNT xx:'logical id'/'vid'/WRITE/NOWRITE/DEN:nn/PAR:nn
DEVICE?
```

If a specific volume is involved, its identification appears as text replacing 'vid.' This text is the visual identification used to distinguish the volume.

Table A–16 lists the specific device type given by xx:. edt

Table A–16: BATCH Device Type Designators

Designator	Meaning
CD	CD11 punched card reader
CR	CR11 high speed punched card reader or the CM11 marked-sense card reader
DB	RP04, RP05 or RP06 disk pack drive
DK	RK05 or RK05F disk cartridge drive
DM	RK06/RK07 disk cartridge drive
DL	RL01/RL02 disk pack drive
DR	RM02/RM03/RM05/RM80 disk pack drive
DT	TU56 DECtape drive
DU	All MSCP disks: RA60, RA80, RA81, RA82, RD31, RD32, RD51, RD52, RD53, RD54, RC25, RX33, RX50
DX	RX01/RX02 floppy disk drive
LP	Any line printer
MM	TU16, TE16, TU45, or TU77 magnetic tape drive
MS	TS11, TSV05, TK25, or TU80 magnetic tape drive
MT	TM11, TU10, TE10, or TS03 magnetic tape drive
MU	TK50 or TU81 magnetic tape drive
TT	Any keyboard line

If the text of the action request indicates a specific volume, the operator should mount the medium on a free unit of the device type specified by xx:. If a volume is not involved, the operator should make sure a unit of the type specified by xx: is ready and on line.

When the unit is ready, the operator must type the device specification (device type and unit number followed by colon) as text in the ANSWER command for that action request number. BATCH assigns that unit and continues processing the user's queued request.

When BATCH is done with the device, it generates an action request to dismount the volume or device. If the request involves a volume, the operator must remove the volume from the device specified and place the unit off line. To continue the BATCH processing, the operator types the CONTINUE command as text in the ANSWER command to OPSER for that action request number. This response tells BATCH to deassign the unit and to continue processing.

A.6 Operator Communication Program: PLEASE

The PLEASE program communicates directly with the operator services program OPSER and, when OPSER is not running, sends text to the system console terminal (KB0:). The program PLEASE is stored in the OPSER package account OPSER$: with protection code of <232>.

Any user can run PLEASE, but only users with SWCFG privilege or users defined as valid operators in the operator table can issue OPSER commands.

You, as the system manager or assigned operator, can run PLEASE to send commands to OPSER. Users who do not have SWCFG privilege and are not valid operators may run PLEASE to send text to the Operator Services Console.

NOTE

> If you want to restrict PLEASE to only be run by users with WREAD privilege, change its protection code to <124>.

A.6.1 Running and Terminating PLEASE

To run PLEASE, type the following command:

```
$ RUN OPSER$:PLEASE
PLEASE V9  RSTS V9  TIMESHARING
#
```

PLEASE prints its identification line and the number sign (#) character as a prompt to indicate it is ready to accept text.

PLEASE accepts one line of text terminated with the Return key. A line of text exceeding the maximum length (255 characters) causes the program to print the error message ?Line too long and reprint the prompt. If the first character of text is not a slash (/) character, PLEASE tries to send the text as a message to OPSER. If the first character of text is a slash, PLEASE tries to send the text as a command to OPSER. To show the operator that it is sending the message or command, PLEASE prints one of the following messages:

```
Message sent to 'OPSER'
Command sent to 'OPSER'
```

The program prints the # prompt again, after which the operator can type another line of text. Entering Ctrl/Z in response to the prompt terminates the program.

When OPSER is not active, PLEASE broadcasts messages on the system console terminal; however, it does not send commands. In these situations, the program notifies the user with one of two messages:

```
%OPSER not active - message broadcast to KB0:
%OPSER not active - command not sent
```

The message printed on keyboard unit 0 is preceded by the text:

```
%MSG for 'OPSER' but it is not active:
```

A.6.2 OPSER Commands through PLEASE

Any user with SWCFG privilege or defined as a valid operator can communicate with OPSER and with jobs on line to OPSER by typing a command through PLEASE. The user denotes an OPSER command by typing the slash (/) character as the first character in the line of text.

Table A–17 summarizes these PLEASE commands to OPSER. Note that the circumflex (^) character marks the location of a required space.

Table A–17: PLEASE Commands to OPSER

Command Syntax	Meaning
/CHA^KBn:	Changes operator services console to keyboard unit n.
/DEL^msgnumber	Deletes outstanding action request.
/DEL^#nn:n	Deletes n oldest action requests for job number #nn.
/DET	Detaches OPSER from the OSC.
/EXI	Terminates OPSER.
/INT^rcvrid:text	Sends unsolicited text to online job.
/LIS^JO	Prints online job table.
/LIS^OP	Prints valid operator table.
/LOG^file;msglevel	Creates log file and sets message level.
/LOG^;msglevel	Changes message level of current log file.
/LOG	Closes current log file and stops recording.
/MES^msglevel	Sets message level for OSC.
/OPE^KBn:[nnn,nnn]	Adds keyboard and account combination to valid operator table.
/OPE^-KBn:[nnn,nnn]	Removes keyboard and account combination from valid operator table.
/RET^msgnumber	Prints associated message.
/RET^m:n	Prints oldest m action requests for job number #n.

When OPSER receives a command from PLEASE, it makes sure that the user has SWCFG privilege or that the user's keyboard and account numbers are present in the valid operator table. If not, OPSER broadcasts the error message ?Invalid oper on the terminal from which the command was made.

Commands sent by a valid operator can be executed by OPSER itself or by a program on line to OPSER. If OPSER itself executes the command, any response from the command is broadcast on the originating terminal. For example, because OPSER itself processes the LIST JOBS command, the online job list generated appears on the originating terminal.

An INTERRUPT command, however, passes text to another program. Any response from the other program is displayed on the OSC and not on the originating terminal. To see a message or response from an INTERRUPT command, the operator should change the OSC to the current keyboard (the line on which PLEASE is running) before transmitting the command. Action requests and messages (responses to commands or text sent to OPSER) are displayed only if the current message level allows.

A.6.3 PLEASE as a CCL Command

If you install PLEASE as a CCL command, the operator can type PLEASE followed by a text line to send commands to OPSER. PLE is the standard abbreviation for the CCL command.

A.7 Terminating Operator Services and Spooling

Usually you terminate OPSER and programs on line to OPSER by running the SHUTUP program. SHUTUP, however, terminates time-sharing operations. If it is necessary to terminate operator services and spooling without shutting the system down, the operator can type a sequence of commands to stop spooling operations in an orderly fashion. To understand the sequence, you must know about OPSER shutdown levels.

A.7.1 OPSER Shutdown Levels

When OPSER receives a command from SHUTUP to perform an orderly shutdown in logical end mode, OPSER sends an END command to each online job in a fixed, logical sequence. OPSER selects by shutdown level the order in which jobs are ended. Online jobs at the lowest level terminate first. When all jobs at one level have terminated and have been removed from OPSER'S internal tables, OPSER begins sending END commands to jobs at the next highest level. When all online jobs have properly terminated, OPSER kills itself and SHUTUP proceeds with the system shutdown.

A.7.2 OPSER Manual Shutdown Procedure

You may need to shut down the SPOOL program without shutting down the entire system. The two possible procedures are as follows:

- The operator can issue the END command to the spooler at any time. This command causes the spooler to shut itself down the next time it is ready to get a new job. Thus, if the spooler is printing a job, it completes the job's output before shutting down. If SPOOL is waiting for a job, it shuts down immediately. The operator can use the ABORT command to terminate the job if the operator wants the job currently printing to end.

  For example, to manually shut down the LP0SPL spooler in logical end mode, use the command:

  ```
  PLE/INT LP0SPL:END
  ```

At the completion of the current job, the spooler kills itself. This shutdown procedure merely mimics what OPSER does on command from SHUTUP. The operator sends the END command, in turn, to jobs at each shutdown level. The output of the LIST JOBS command gives the shutdown levels for each online job. When OPSER displays END messages for all jobs at one shutdown level, the operator can send the END command to all jobs at the next highest shutdown level. After the online job table is clear and OPSER has generated END messages for all jobs, the operator can send the EXIT command to OPSER.

- A second way of shutting down the spooler is with an OFFLINE command. The OFFLINE command immediately stops any job that is printing and prints an ABORT message on the listing to indicate the action taken. The OFFLINE command also causes SPOOL to shut down after the current job finishes.

  For example, to manually shut down the BA0SPL job immediately, use the command:

  ```
  PLE/INT BA0SPL:OFFLINE
  ```

  BA0SPL aborts the current job, clears it from QUEUE.SYS, and kills itself.

If the operator wants to terminate a printing job for shutdown or any other reason, the line printer must be on line. Under no circumstances does the spooler consider a job completed until all the data that the spooler tries to print (including termination messages) is actually printed. If the operator aborts the job with the ABORT command, the spooler deletes (at the first opportunity) whatever data is already buffered; the line printer must be on line to allow this.

If the operator wants no further output, use the REMOVE/JOB command to remove the spooler job. In this case, no clean-up can be performed. This process may be necessary when certain hardware fails (for example, if the line printer interface does not raise the READY flag). It should only be used in a crisis. The job that was being printed should be killed from the queue file by a QUE/K command from the operator.

A.8 RESTOR as an OPSER Controlled Program

The RESTOR program runs as a job on line to OPSER only after it has finally detached from its terminal. When RESTOR is running detached, the operator can send commands to it through OPSER. If, after having been detached, RESTOR is later attached to a terminal again, the operator can communicate with RESTOR either at its terminal or through the OPSER program.

Table A–18 summarizes the commands that RESTOR recognizes. Note that the circumflex (^) character marks the location of a required space.

Table A–18: RESTOR Commands through OPSER

Command Syntax	Meaning
ABO[RT]	Terminates the RESTOR run immediately and removes RESTOR from OPSER internal tables.
CON[TINUE]	Wakes up the RESTOR program to continue processing after a PAUSE command or continues processing after an operator has performed some requested action.
DET[ACH]	Valid only under these conditions: • If RESTOR is attached • OPSER is running • The RESTOR listing file is not on the terminal to which RESTOR is attached • No messages are currently pending for the specific RESTOR job If these conditions are met, RESTOR detaches and is given a receiver identification of BACKnn where nn is its job number. All further interaction with RESTOR is done through OPSER.
END	Finishes transferring the current file and terminates processing.
LAS[T]	At the OSC, prints the most recent message generated by this RESTOR job.
LEG[AL]	At the OSC, prints a list of commands that can legally be given for this phase of the RESTOR run.
NOT[ICE]^text	Inserts the specified text in the RESTOR listing file. Precedes the text with the heading NOTICE FROM OPERATOR.
PAU[SE] STA[TUS]	Places RESTOR in a sleep state, during which it responds to most commands, continues normal processing in response to a CONTINUE command, and terminates processing on an ABORT command. Prints a status report for this RESTOR job.

While RESTOR is running detached, it generates all of its requests for operator interaction through OPSER as messages. The operator uses the INTERRUPT command to respond to such requests. See the *RSTS/E Utilities Reference Manual* for the proper responses to RESTOR requests.

Appendix B

Number Conversion

Many applications require a number based on bit values in a PDP-11 word. Table B–1 lists the octal and decimal values for each bit in the PDP-11 word.

Table B–1: Bit Values in the PDP-11 Word

Bit Number	Octal Value	Decimal Value
0	1	1
1	2	2
2	4	4
3	10	8
4	20	16
5	40	32
6	100	64
7	200	128
8	400	256
9	1000	512
10	2000	1024
11	4000	2048
12	10000	4096
13	20000	8192
14	40000	16384
15	100000	32768

Appendix C

Disk Device Sizes

Table C–1 lists the device cluster size and device size (in 512-byte blocks) for each disk that RSTS/E supports. All values are in decimal.

Table C–1: Disk Device Sizes

Disk Type	Minimum Device Cluster Size	Total Available Size [1] (Blocks)
RX50	1	800
RX33	1	2400
RK05	1	4800
RK05F	1	4800 [2]
RL01	1	10220
RL02	1	20457
RD51	1	21599
RD52	1	60479
RC25	1	50901 [2]
RK06	1	27104
RD31	1	41559
RK07	1	53761
RP02	2	39998
RP03	2	79998
RD32	2	83204
RM02	4	131648
RM03	4	131648
RD53	4	138668
RP04	4	171796
RP05	4	171796
RA80	4	237208
RM80	4	242572
RD54	8	311200
RP06	8	340664
RA60	8	400168
RM05	8	500352
RA81	16	891056
RA82	32	1216640

[1] Size when disk is initialized at lowest acceptable cluster size

[2] For each unit; 2 units per drive

Appendix D

General RSTS/E Messages

This appendix lists many of the general messages that you may encounter when using RSTS/E. It is not a complete listing. Table D–1 also indicates the probable cause of the error in your use of a command or system program.

If you perform DECnet/E operations, you may receive messages that are described in DECnet/E documentation but not in this appendix. In addition, the utility programs that support DCL can produce messages not all of which are listed here. Those messages are listed in the *RSTS/E Utilities Reference Manual*.

D.1 Special Characters Used in Error Messages

The question mark (?), double question mark (??), and percent sign (%) characters preceding a message indicate its severity. A single question mark precedes standard error messages, which means that the command failed to do what you asked but you can continue. For example:

```
?Too many parameters
```

In this case, the command failed because you included too many parameters.

A double question mark precedes severe error messages, which means that the command failed to do what you asked and you cannot continue. For example:

```
??Fatal system I/O failure
```

By contrast, the percent sign identifies a warning message, which means that the command may not have worked as you intended. For example:

```
%Logical name has not been assigned
```

In this case, the MOUNT command worked but no logical name was assigned to the device.

Informational messages do not have a preceding character; they provide extra details about the effects of a command. For example:

```
Queue file being reorganized - please wait...
```

This message informs you that, as a result of your command, the file is being reorganized.

Angle brackets (< >) surrounding text indicate a place holder for what the system inserts when an error message occurs.

Table D–1 lists the messages that you can get when using RSTS/E and the probable causes.

Table D–1: General RSTS/E Messages

Message	Meaning
?Account or device in use	The device cannot be mounted or dismounted because it is open or has one or more open files. This error can also occur when a user attempts to access a non-shared disk from a job other than the one specified in the /NOSHARE=n command.
?Additional argument required	You did not specify enough arguments in the DCL function.
?Additional qualifier required	You did not include a qualifier required in a command.
%All volumes must be of the same media type	You attempted to write a Backup set on both disk and tape.
?Ambiguous keyword	You abbreviated a keyword to short; DCL cannot distinguish it from other keywords.
%An unrecognized character was typed - try again	There were extra characters in the command line that BACKUP did not recognize as meaningful.
?Argument not allowed	You used an argument with a qualifier that does not accept one.
?Argument required	An argument was not supplied with a qualifier which required one. For example, /SINCE=, without an argument, would generate this error. The user did not supply an argument to a DCL function that required one.
??Bad directory for device	The directory of the device referenced is in an unreadable format. For example, the tape format differs from the format you specified with the MOUNT command.
%Block header CRC error detected for block	Although this block was read successfully by the source device, BACKUP discovered that there was an error in the block. BACKUP will attempt to recover the data using recovery data if possible.
?Blocksize must be a multiple of <16 or 512>	An invalid blocksize was specified. Blocksize must be a multiple of 16 for tape and 512 for disk.
?Can't access terminal<error text>	An error occurred while attempting to access the terminal to send the setup file using the SET TERMINAL command.
?Can't find file or account	Either the account or file does not exist, or you typed a file specification incorrectly.

(continued on next page)

Message **Meaning**

?Can't mount a private disk as public

You tried to mount as public (using the MOUNT command's /PUBLIC qualifier) a disk that was initialized as private.

?Can't rebuild disk because device is write protected

A user with MOUNT privilege attempted to rebuild a disk (using the MOUNT command's /REBUILD qualifier), and the drive is write protected Therefore, the mount and rebuild operations fail.

?Command not installed

You typed a DCL command that has not been installed on your system. The DCL keyboard monitor could not find the utility needed to carry out your command. This message can also be displayed if you type a remote fi specification in a command that allows remote file specifications, but your system does not have DECnet/E.

?Command requires<privilege> privilege

The user does not have the appropriate privileges to complete an operatior

?Command too long

The length of the command, including continuation lines, exceeds the maximum allowed for the command. Or, the length of the translated command string exceeds the maximum allowed for the command. For example, this message occurs with the LINK command in either of two cases:

- The text you typed in response to the $ prompt and the Files: or Root files: prompt added up to more than 127 characters after translation.
- With LINK/COBOL, the text was longer than 80 characters after translation.

%Command will proceed as requested

The user's symbol tables could not be written out to the DCL work file, bu the command will be processed. The user may be over quota. Corrective action should be taken before the user continues processing.

?Comma required

A DCL function required two or more arguments and they were not separated by a comma.

?Conflicting arguments

Multiple arguments to a qualifier conflict. For example, both an AFTER date and an exact date were specified to the /CREATED qualifier.

?Conflicting elements

You specified qualifiers or parameters in a command that do not agree in type (for example, a print qualifier with a batch server).

?CPU limit exceeds queue's maximum

You specified a /CPU_LIMIT value in a SUBMIT command larger than the maximum allowed for the queue.

?Data error on device

One or more characters may have been transmitted incorrectly due to a parity error, bad punch combination on a card, or similar error.

(continued on next pag(

Table D–1 (Cont.): General RSTS/E Messages

Message **Meaning**

?Data error or incorrect density

This message occurs for one of the following reasons:

- You specified the incorrect density.
- The data on the tape is damaged.
- The tape drive is faulty.

?Default macro for KBn: does not exist

The /INQUIRE command could not recognize the terminal and there was no default macro specified for this keyboard in the default file, TERDFL.SYS. You should supply a default macro for the keyboard. Until that is done, the keyboard's characteristics have to be set manually.

Density of <density> BPI not available, using <density> BPI instead

The requested density was not available on the drive. BACKUP reports the density that is being used.

?Device does not exist

The device name you specified does not exist on your system.

%Device hung or write locked

%Dismount will proceed as requested

You attempted to dismount a disk that has been physically dismounted. However, the logical dismount will succeed.

?Device hung or write locked

Check the hardware condition of the device requested. Possible causes of this error include:

- A line printer out of paper
- A disk drive being off line
- A DB or DR disk is dual-ported and is not write-protected

%Device in use

You specified /NOSHAREABLE in an INITIALIZE/SERVER command; however, the print server's device is not available or no pseudo-keyboards are currently available for the batch server. PBS allocates the device as soon as it becomes available.

?Device is not a terminal

A device other than a terminal was specified. Only specify keyboards.

?Device is restricted

This message occurs when a user without DEVICE privilege attempts to open files on a disk restricted to users with DEVICE privilege.

?Device must be disk

You specified a file on a nondisk device, where a disk device is required.

?Device must be disk or tape

The device specified was not a disk or tape, and only disk or tape is valid.

?Device name may not be specified

A device name is not valid at this point, but one was specified.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message **Meaning**

?Device not available

The specified device exists on the system, but an attempt to **allocate** or use it is prohibited for one of the following reasons:

- The device is currently reserved by another job.
- The device requires privilege for ownership and you do have the necessary privilege.
- The device or its controller has been disabled by the manager.
- The device is a keyboard line for pseudo keyboard use only.

?Device not file-structured

An attempt was made to access a device, other than a disk drive or magneti tape drive, as a file-structured device.

?Device not write protected

You specified the /NOWRITE qualifier when you tried to mount a magnetic tape, but the device is not write protected. Write protect the device by removing the plastic ring from the magnetic tape hub.

?Device offline

You tried to use a tape or disk, but the device is off line.

%Device write protected

The magnetic tape or disk is protected against write access. Therefore, you can only read files on the device, but cannot write (perform output) to the device. If you want to write to the device, first write enable the device, next dismount the device, and then mount the device again.

?Device write protected

You requested write access to a device that is write protected. Write enable the device and then retype the command.

%Directory attributes verification failure for directory <directory>

The directory attributes stored in the Backup set are different from the directory attributes as stored in the directory structure. This is a verification mismatch. This can happen if a user logs in or out during a backup.

?Directory does not exist

A file specification indicates a directory that does not exist on the particular private disk. Or, a wildcard directory specification failed to produce a match on the disk specified. This error can occur only with private disks.

??Disk error during swap

A hardware error occurs when your job is swapped into or out of memory. The contents of the job area are lost, but the job remains logged in to the system and returns to DCL. Report such occurrences to the system manager.

Disk is being rebuilt - wait...

This informational message is displayed when a user attempts to rebuild a disk. The disk is logically mounted when the rebuilding operation is complete.

?Disk is mounted non-shared

A user attempted to dismount a disk that was mounted as nonshared (by a different job) by specifying /PUBLIC in the DISMOUNT command.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Disk is mounted private or non-shared	A user attempted to dismount a disk that was mounted as nonshared (by the same job) or private by specifying /PUBLIC in the DISMOUNT command.
?Disk is mounted public	A user with MOUNT privilege attempted to dismount a public disk that was mounted as public, without specifying /PUBLIC in the DISMOUNT command, so the dismount fails.
%Disk is mounted read-only	This warning occurs if you do not specify either read or write access (/NOWRITE or /WRITE) when mounting a disk initialized as read-only. Therefore, you are granted read access only.
?Disk is NOSHARE to another job; can't verify pack-id	A user with MOUNT privilege attempted to mount a disk that is mounted /NOSHARE to another job.
%Disk is restricted and mounted non-shared, read-only	A user specified /NOREBUILD (to suppress rebuilding) and /NOSHARE in the MOUNT command and the disk was dirty. Therefore, the disk is mounted nonshared and restricted, and read-only access is granted. ("Restricted" means that access is allowed only to users with DEVICE privilege.)
%Disk is restricted and mounted private, read-only	A user specified /NOREBUILD (to suppress rebuilding) and /SHARE in the MOUNT command and the disk was dirty. Therefore, the disk is mounted nonshared and restricted, and read-only access is granted. ("Restricted" means that access is allowed only to users with DEVICE privilege.)
?Disk needs rebuilding but device is write protected	A user with MOUNT privilege attempted to mount a dirty disk (and did not specify /REBUILD or /NOREBUILD), but the device is write protected and the automatic rebuilding cannot be performed. Therefore, the mount fails.
?Disk needs rebuilding but you do not have MOUNT privilege	This message occurs when a user without MOUNT privilege tries to mount a private disk that was not logically dismounted. The system manager can correct the situation by rebuilding the disk.
?Disk needs rebuilding, so it can not be NORESTRICT	This message occurs when a user with MOUNT privilege tries to mount a dirty disk with both /NOREBUILD and /NORESTRICT qualifiers.
?Disk pack is not mounted	The DISMOUNT command was attempted, but the disk pack was not mounted on the specified disk drive.
??Disk pack mount error	Fatal disk mounting error. The disk is corrupt and cannot be successfully mounted with the MOUNT command.
?Dismount will proceed as requested	The disk was physically dismounted before the logical dismount was finished. RSTS/E will flag the device as dismounted anyway.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Division by zero	The expression specified by the user attempted to divide by zero.
?Do not specify file name or type	This message can occur with the first parameter of the ASSIGN command. Remember to use a space between the first parameter (the string you assign) and the second parameter (the name you assign it). For example, "ASSIGN DR3: X", not "ASSIGN DR3:X".
%Dual-ported disk, may still be mounted on another system	You specified a DB or DR disk in a DISMOUNT command that is dual-ported. So, it may still be mounted on another system.
?Equal sign required	You used a qualifier that requires an argument, but you did not give an argument.
?Error assigning Backup set device	The RSTS/E monitor reported an error in response to BACKUP's request to assign the Backup set device. This message will be followed by another message describing the problem.
?Error closing Backup set	The RSTS/E monitor reported an error in response to BACKUP's request to close the Backup set. A message detailing the problem will follow.
?Error creating directory<directory>	The RSTS/E monitor reported an error in response to BACKUP's request to create a directory. A message detailing the problem will follow.
?Error creating disk initialization command file	The RSTS/E monitor reported an error in response to BACKUP's request to create the command file used to initialize disks. A message detailing the problem will follow.
?Error creating dynamic region	The RSTS/E monitor reported an error in response to BACKUP's request to create a dynamic region. A message detailing the problem will follow.
?Error creating job to initialize disk	The RSTS/E monitor reported an error in response to BACKUP's request to create a job, used to initialize disks. A message detailing the problem will follow.
?Error creating output directory	The RSTS/E monitor reported an error in response to BACKUP's request to create a directory. A message detailing the problem will follow.
?Error deassigning Backup set device	The RSTS/E monitor reported an error in response to BACKUP's request to deassign the Backup set device. A message detailing the problem will follow.
?Error dismounting disk to initialize it	The RSTS/E monitor reported an error in response to BACKUP's request to dismount a disk so that it can be reinitialized. A message detailing the problem will follow.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message **Meaning**

?Error in forms definition<forms-name>

<error message>

An error, as the error message in the second line describes, was discovered in the forms definition displayed.

?Error initializing Backup set

The RSTS/E monitor reported an error in response to BACKUP's request to initialize the Backup set. This message will be followed by another message detailing the problem.

?Error mounting output disk

This message indicates that the monitor has reported an error as a result of BACKUP's mount request. It will be followed by an error detailing the problem.

?Error mounting Backup set

The RSTS/E monitor reported an error in response to BACKUP's request to mount the Backup set. A message detailing the problem will follow.

?Error number nn

An I/O error occurred while the system was attempting to retrieve an error message. Possible causes could be that the device containing the system error file (ERR.SYS) is off line, or that the system error file contains a bad block.

This is a serious error, and should be reported to the system manager.

?Error obtaining file flags for file <filename> for verification

The RSTS/E monitor reported an error when BACKUP requested to find out what the file flags for a file to verify were set to. This message will be followed by another message describing the problem.

?Error opening disk non-file-structured to initialize it

<error message>

This message occurs when BACKUP is requested to initialize a disk and the RSTS/E monitor is unable to open it in non-file-structured mode. This generally happens because a disk unit is off line or no disk is mounted in the drive. This can also occur for other reasons, such as no access to non-file-structured disks or other device specific errors.

?Error opening file <filename> for input - skipping rest of directory

The RSTS/E monitor reported an error in response to BACKUP's request to open a file in the directory (note that since the filename is unknown at the time, this may contain wildcards). A message detailing the problem will follow. Since BACKUP scans the directory sequentially, the rest of the directory will be skipped.

?Error opening file <filename> for output

The monitor reported an error to BACKUP when it was trying to create a file. This message will be followed by a message describing the problem.

?Error opening file <filename> for verification

The RSTS/E monitor reported an error in response to BACKUP's request to open a file for verification. This message will be followed by another message detailing the problem.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Error opening PBS file <file-spec> **<error message>**	An error, as the error message in the second line describes, occurred opening the PBS file indicated.
?Error opening Backup set	The RSTS/E monitor reported an error in response to BACKUP's request to open the Backup set. Another message will follow, describing the problem.
?Error opening Backup set non-file structured	The RSTS/E monitor reported an error in response to BACKUP's request to open the Backup set device for control purposes. An error follows which will detail the problem.
?Error performing PPN lookup	The RSTS/E monitor reported an error in response to BACKUP's request to look up PPNs to back up. A message detailing the problem will follow.
?Error reading accounting data for directory <directory>	The RSTS/E monitor reported an error in response to BACKUP's request to read the accounting data for a directory. A message detailing the problem will follow.
?Error reading directory attributes for directory <directory>	The RSTS/E monitor reported an error in response to BACKUP's request to read directory attributes. A message detailing the problem will follow.
?Error reading file attributes for file <filename>	The RSTS/E monitor reported an error in response to BACKUP's request to read file attributes for a file being backed up. A message detailing the problem will follow.
?Error reading file <filename> block <block>	The RSTS/E monitor reported an error in response to BACKUP's request to read file data. A message detailing the problem will follow.
?Error reading output volume label	The monitor reported an error to BACKUP in response to a request to determine the volume label of the output device. This message will be followed by a message detailing the problem.
?Error reading Backup set	This message indicates that an error was reported by the monitor to BACKUP when attempting to read the Backup set. It will be followed by a device specific error message describing the problem. In addition, it may be followed by other messages if BACKUP is unable to recover from the lost data.
?Error reading Backup set attributes	The RSTS/E monitor reported an error in response to BACKUP's request to read the Backup set attributes. An error follows which will detail the

(continued on next page)

Table D-1 (Cont.): General RSTS/E Messages

Message	Meaning
?Error reading Backup set volume label	The RSTS/E monitor reported an error in response to BACKUP's request to read the Backup set label. A message detailing the problem follows.
?Error restoring file dates for file <filename>	The RSTS/E monitor reported an error in response to BACKUP's attempt to restore file dates. Another message describing the problem follows this message.
?Error rewinding Backup set	BACKUP encountered an error when asking the monitor to rewind the Backup set. A message detailing the problem follows.
?Error setting end-of-volume flag	The RSTS/E monitor reported an error in response to BACKUP's request to set the end-of-volume flag. A message detailing the problem follows.
?Error setting file flags for file <filename>	The RSTS/E monitor reported an error in response to BACKUP's attempt to set file flags (caching, protected, etc.). Another message describing the problem follows this message.
?Error setting RTS name for file <filename>	The RSTS/E monitor reported an error in response to BACKUP's attempt to set the runtime system name for a file. Another message describing the problem follows this message.
<error text> opening setup file <filename>	An error occurred while trying to gain access to the setup file.
<error text> opening speed file	An error occurred while trying to gain access to the speed file. The speed is not set and the rest of the SET TERMINAL command is executed.
<error text> opening/reading default file	An error occurred while processing the default file.
<error text> reading setup file <filename>	An error occurred while processing the setup file.
<error text> reading speed file	An error occurred while processing the speed file. The speed is not set and the rest of the SET TERMINAL command is executed.
?Error verifying directory attributes for directory <directory>	The RSTS/E monitor reported an error in response to BACKUP's request to read directory attributes for verification. Another message describing the problem follows this message.
?Error verifying directory <directory>	The RSTS/E monitor reported an error in response to BACKUP's request to obtain directory information for verification. Another message describing the problem follows this message.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Error writing directory attributes for directory <directory>	The RSTS/E monitor reported an error in response to BACKUP's request to write directory attributes. Another message describing the problem follows this message.
?Error writing file attributes for file <filename>	The monitor reported an error to BACKUP when it was trying to restore file attributes. Another message describing the problem follows this message.
?Error writing Backup set	The RSTS/E monitor reported an error in response to BACKUP's request to write to the Backup set. Another message describing the problem follows this message.
?Error writing Backup set attributes	The RSTS/E monitor reported an error in response to BACKUP's request to write the Backup set attributes. Another message describing the problem follows this message.
?Error writing to disk initialization command file	The RSTS/E monitor reported an error in response to BACKUP's request to write to the command file used to initialize disks.Another message describing the problem follows this message.
?Error writing to list file	The RSTS/E monitor reported an error in response to BACKUP's request to write to the list file. Another message describing the problem follows this message.
?Extra file data found in file <filename> - ignoring it	RESTORE found file data blocks for blocks beyond the written end of file. If this happened with a SAVE set produced by RSTS/E and was not accompanied by other messages, submit a SPR with a copy of the Backup set.
??Fatal system I/O failure	An I/O error has occurred on the system level. The results of the last command are unpredictable. This error is caused by a hardware condition. Report such occurrences to your system manager.
?File attributes for file <filename> may be incorrect	If BACKUP has used recovery data to restore a file, there is a chance that the file attributes may not be correct. If this is a possibility, BACKUP informs the user with this message.
?File data block for file <filename> duplicated in Backup set	RESTORE found a file block duplicated in the Backup set. If this this happened with a SAVE set produced by RSTS/E and was not accompanied by other errors, submit a SPR containing a copy of the Backup set.
?File data block for file <filename> missing in Backup set	RESTORE found a file block missing in the Backup set. If this was not accompanied by other errors, submit a SPR containing a copy of the Backup set.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?File data in file may be incorrect	This message can occur if there are several missing blocks from a Backup set. It should be accompanied by other messages, describing the cause of the problem. If not, submit a SPR with a copy of the Backup set.
?File does not exist	This message occurs for one of the following reasons: • An input file that must be present is not present. • A wildcard file specification does not match any files. • A file is not accessible because of its assigned protection code.
File <filename> not replaced	This is an informational message, which reports that a file was not restored because there was a file of the same name already existing and you did not use the /REPLACE qualifier.
File <filename> restored as non-contiguous	The RSTS/E monitor was unable to locate contiguous space to restore a contiguous file as contiguous.
?File name attribute invalid - skipping file	If this Backup set was written by RSTS/E BACKUP, submit a SPR containing a copy of the Backup set. If this Backup set was written by any other operating system, it indicates that the file name could not be mapped into a RSTS/E name, and was skipped. Rename the file on the source system to something which will fit into the RSTS/E file naming scheme.
?File name attribute missing - skipping file	Submit a SPR with a copy of the Backup set.
?File name required	A file specification you typed does not include a file name, but one is needed.
?File specification required	This error occurs if you specify node:: without a file specification (except with DIRECTORY). It also occurs if you have two commas with no characters between them in a file specification list.
?Files cannot be on different nodes	This message occurs with network operations. Each input file specification you include in a network command must be on the same node.
?Form <form-name> does not exist	The form name you specified was not defined by the system manager. See your system manager to find out what form names are available.
?Forms Definition File does not exist	The forms definition file, PBS$:FORMS.SYS does not exist. See your system manager.
?Forms not defined [for server <server-name>]	The forms you specified could not be found in the forms definition file. See your system manager.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
%ID label ignored	You mounted a tape in DOS format and specified an ID label. Identification labels are not encoded on DOS tapes; therefore, the label you specified is not recognized by the MOUNT command.
%ID label should be specified when you mount an ANSI tape	You mounted a tape in ANSI format, but did not specify the tape identification label. DIGITAL recommends that you specify the identification label. Thus, the MOUNT command can verify that the tape you selected has been mounted.
?ID labels don't match	The identification label you specified does not match the identification label encoded on the tape. To correct the problem, specify the correct identification label. If you do not know what identification label is encoded on the tape, you can omit the ID label from the MOUNT command.
?Illegal byte count for I/O	The range of memory starting at the load address given is not available. Refer to the memory status report of a display program (SYSTAT or DISPLY) to select an available range of memory. Note that you can also use the SHOW MEMORY command.
?Illegal filename	The file name given in the command contains characters other than alphabetic or numeric characters.
?Illegal switch usage	A CCL command contains an error in an otherwise valid CCL switch (qualifier). For example, you cannot use the /SI:n switch without a value for n or a colon; or you cannot specify more than one of the same type of CCL switch.
?Illegal value - n	The swap file number specified is not 0, 1, or 3. (The swap file 2 already exists on the system disk and can not be added.)
?Impossible density for this device	You tried to mount or initialize a tape, but specified a density not available on the tape drive you used. To correct the problem, use another drive or specify a different density.
?Incorrect density	You specified a density different from the tape's density.
?Incorrect density or uninitialized tape	This message can occur for the following reasons: • You tried to mounta tape that had not been initialized. • You specified an incorrect density. If the tape has not been initialized, use the INITIALIZE command. Otherwise, specify the correct density. If the tape has not been initialized, use the INITIALIZE command. Otherwise, specify the correct density.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Invalid account	The account specified was not valid either containing invalid characters (for example, [^^^]) or an invalid format (for example, [1,2345]).
?Invalid argument	This error can occur when you use a qualifier in the form /qualifier=argument. The qualifier you used is spelled properly, and the equal sign (=) or colon (:) is present; however, the argument is either missing or syntactically invalid. RSTS/E displays this error message when there is no more specific message to describe the syntax error.
?Invalid at interactive level	You specified a command that is invalid at the interactive level. (For example, GOTO LABEL)
?Invalid attribute length - attribute skipped	If you get this message, submit an SPR with a copy of the Backup set that generated the error.
?Invalid attribute - attribute skipped	This message indicates that the version number of a record was higher than the highest version known to BACKUP. If this error occurs trying to restore a Backup set written by a newer version of BACKUP on an older version of BACKUP, the operation may not work properly. If the version of BACKUP writing the tape is older or the same as the one doing the RESTORE, submit an SPR with a copy of the Backup set that generated the error.
?Invalid BATCH command	Although the command is valid in interactive mode, the command is not a valid BATCH command because it implies or requires interaction with the user. (For example, INQUIRE, SET NODATA)
?Invalid CCL command	You used the CCL prefix followed by a command that is not installed as a CCL command on your system.
?Invalid channel number	The I/O channel number specified was invalid. Channel numbers can range from 1 to 13 on OPEN, READ and CLOSE commands, and from 0 to 13 on WRITE commands.
?Invalid character	You typed an invalid punctuation character.
?Invalid command	The command name you gave is not a DCL command, and is not defined on your system as a CCL command. Or, the line begins with a punctuation character rather than with a keyword.
?Invalid date	A date either has improper syntax, represents a nonexistent date (for example, 30-Feb), or represents a date before 1970 or after 1999.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Invalid density - n	You tried to mount or initialize a tape, but specified a density other than 800 or 1600 bpi. The n is the density you specified.
?Invalid device	The device specified could not be found in the list of standard devices or in the user-defined macros (TERMAC.SYS). Specify a valid device.
%Invalid directory attribute record - skipping	This message indicates that a directory attribute code found in a Backup set was unknown to BACKUP. If this error occurs trying to restore a Backup set written by a newer version of BACKUP on an older version of BACKUP, the operation may not work properly. If the version of BACKUP writing the tape is older or the same as the one doing the RESTORE, submit an SPR with a copy of the Backup set that generated the error.
?Invalid entry name	The entry name you specified is invalid.
?Invalid expression	This covers a range of expression problems. The most common of these is an expression ending in an operator without supplying a second operand (for example, A = B *).
?Invalid file specification	A local file specification has improper syntax.
?Invalid fill factor	A fill factor was specified that was less than 0 or greater than 6.
?Invalid form definition	The definition of the specified form contains an invalid keyword or keyword argument.
?Invalid form name	The form name you specified is either longer than six characters or consists of one or more nonalphanumeric characters.
?Invalid function	The function name specified after F$ is invalid. The minimum abbreviation point is not met for the function (for example, F$LE is not valid because of F$LEFT and F$LEN).
%Invalid keyboard numbers n , n in default file	The numbers you specified in the default file were less than 0, greater than 127, or the first number of the range is greater than the second number of the range.
%Invalid keyboard numbers n , n in speed file	The numbers you specified in the speed file were less than 0, greater than 127, or the first number of the range is greater than the second number of the range.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Invalid keyword	The keyword is not recognized. This error occurs with keywords that are not qualifiers and are not command names. (For example, it can occur with the options of SET and SHOW, and with qualifier values that are keywords.)
?Invalid label	The label specified in the command procedure contains invalid characters.
?Invalid node name	The node name field of the file specification contained invalid characters.
?Invalid operation on system disk	You specified an operation on the system disk which is invalid (for example, DISMOUNT).
?Invalid operator	The operator you specified in the expression is not a defined logical, arithmetic, string, arithmetic comparison or string comparison operator.
?Invalid PPN	The project-programmer number (PPN) you specified does not have valid syntax.
?Invalid print device	The device you specified is not a valid print device because it is not a line printer or terminal.
?Invalid private delimiter	The delimiter you specified was not in one of the correct formats.
?Invalid qualifier	The qualifier keyword is not valid in the command you typed. (This message may indicate an error in spelling or typing.)
?Invalid qualifier for disk	This message occurs when you specify an invalid qualifier for disks. For example, /FORMAT=ANSI (applies only to magnetic tapes).
?Invalid qualifier for tape	This message occurs when you specify an invalid qualifier for tapes. For example, /PRIVATE (applies only to disks).
?Invalid queue name	The queue name you specified is invalid.
?Invalid record length in Backup set - record skipped	If you get this message, submit an SPR with a copy of the Backup set that generated the error.
%Invalid Backup set attributes - can't use it	The Backup set does not have the attributes correct for a BACKUP Backup set.

(continued on next page)

Table D-1 (Cont.): General RSTS/E Messages

Message	Meaning
?Invalid server name	The server name you specified is invalid.
?Invalid speed	The speed specified was not a valid speed for the interface of the terminal. Only specify speeds that the interface supports.
?Invalid symbol name	You specified a symbol name that contains invalid characters.
?Invalid time	A time either has improper syntax or represents a nonexistent time (like 25:00 or 13:00PM).
?Invalid width	You specified a width that was less than 1 or greater than 254.
?Invalid with network file specification	You gave a network file specification in one of the commands that accepts them (RENAME, COPY, and so on), but you also gave a qualifier that can be used only with local operations.
?I/O to detached keyboard	This message can result from one of two actions: • You tried to perform I/O with a terminal line that is used for dial-up terminals, but nobody was dialed in. • Your job became detached (perhaps because you were dialed in and your line was later hung up) and then tried to perform I/O with the terminal. The second situation either causes the job to hibernate or causes this error condition, after which the job hibernates. You see this message when you subsequently attach to the job.
?Keyword required	You typed nonalphanumeric characters when a keyword is needed instead. (If you type alphanumeric characters without a valid keyword, you receive the error message, ?Invalid keyword.)
?Label already defined	A label was defined more than once in an indirect command procedure.
?Label too long	The label you specified in the indirect command procedure was longer than 255 characters.
?Log file already open	A log file was already currently open when you attempted to open another log file using the OPEN/LOG_FILE command.
?Log file not currently open	You attempted to enable or disable logging using the SET LOG_FILE command, and a log file did not exist.
?Log file print queue closed	The print queue required for a SUBMIT command's /LOG_QUEUE qualifier is closed or marked for deletion. Use a different print queue for the log file.

(continued on next page)

Message **Meaning**

?Log file print queue does not exist

The print queue you specified with the /LOG_QUEUE qualifier in a SUBMIT command does not exist. Specify a different queue.

%Logical name has not been assigned

This warning can be displayed for one of two reasons:

- You had INSTAL privilege and specified a logical name that was not assigned as a system logical. (This means that neither the alternate logical name nor the pack-id label were assigned.)
- You did not have INSTAL privilege and specified an alternate logical name when attempting to mount a disk.

The mount succeeds in both cases, but the logical name is not assigned.

?Logins are disabled—please try again later

This message can be displayed when you try to log in, for one of two reasons:

- The system is full, so it cannot accept additional users.
- The system manager has disabled logins.

(Possibly, logins are disabled because the system will be shut down shortly.)

?Magtape record length error

When performing input from magnetic tape, the record on tape was found to be longer than the buffer designated to handle the record.

?Magtape select error

When access to a magnetic tape drive was attempted, the selected unit was found to be off line. This error can occur when you transfer data to or from a tape.

?Map or executable file required

With LINK, you specified /NOEXECUTABLE and did not specify a map file.

??Maximum memory exceeded

This is a nonrecoverable RSTS/E error caused by the following conditions:

- While loading a program into memory, the job's private maximum memory size was reached.
- While executing a program, the system required more memory for string or I/O buffer space, and the job's private maximum memory size or the system maximum was reached.

?Missing closing apostrophe

The user did not specify matching apostrophes ('... ') when requesting apostrophe substitution.

?Missing closing bracket

This error can occur in a local or remote file specification. There is a left bracket ([) or left angle bracket (<), but no right bracket (]) or right angle bracket (>).

(continued on next page)

Table D-1 (Cont.): General RSTS/E Messages

Message	Meaning
?Missing closing quote	A quotation mark (") is not matched with another quotation mark. (This error can occur in remote file specifications.)
?Missing device or file name	The command must contain either a device specification or a device and file name specification. If you use the /SI:n switch, a file name must be present.
?Missing open parenthesis	You did not specify an open parenthesis [(] when one was expected.
%More than 16 speeds specified for <terminal>	While processing the TERSPD.SYS file, a terminal was found that had mor than 16 speeds specified. Only the first 16 speeds are used.
?MOUNT privilege is required	You attempted an operation that requires MOUNT privilege. For example, a user without MOUNT privilege can mount only private disks.
?MOUNT privilege required to dismount a public disk	A user without MOUNT privilege tried to dismount a disk initialized as public. A user without MOUNT privilege can dismount only private disks.
?MOUNT privilege required to rebuild a disk	This message occurs when a user without MOUNT privilege attempts to rebuild a private disk with the /REBUILD qualifier of the MOUNT command. Have your system manager rebuild the disk.
?Name or account now exists	You attempted either to COPY to or RENAME an existing file. This error can occur with RENAME if you do not specify /REPLACE and the output file already exists. It can also occur with COPY if you specify /NOREPLAC and the output file already exists.
?Network node names must be the same	Different node names were specified on input file specifications.
?No buffer space available	The system is overloaded and cannot complete your command because smal buffers are currently unavailable. Try the command again later.
?No channels available	An I/O or DCL command file channel is not available. You may have issue the at (@) command or an F$SEARCH or OPEN command, but a channel i not available.
?No default {Print,Batch} queue	You did not specify a queue name with the PRINT or SUBMIT command, and no default queue exists.
?No default print queue for log file	The /LOG_QUEUE qualifier in a SUBMIT command was specified without an explicit queue name; however, no default print queue exists. Use an explicit queue name for the log file.

(continued on next page

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?No file name or type permitted	A device:ppn syntax was expected and you supplied a full file specification which included either a file name or type.
?Non-executable file	This error occurs if the file you are trying to run is a source file; for example, .BAS. You need to compile and link the file before you run it. (Note that an executable file includes the value 64 in its protection code.)
?Non-printable character	You typed a control character.
??Non-res run-time system	This message generally indicates hardware problems. For example, the run-time system referenced has not been loaded into memory and cannot be loaded for some reason, and is therefore nonresident.
%No owner rights to {print,batch} entry <entry-spec>	The entry shown is not your own and you do not have sufficient privilege to delete it or modify it.
?NO prefix not allowed	You used the /NO prefix improperly.
%No {Print,Batch} entry <entry-spec> found	No entries matching the entry-spec you specified in a SHOW ENTRY or SET ENTRY command were found.
%No {Print,Batch} queue <queue-name> found	No queues matching the queue name you specified in a SHOW QUEUE or SET QUEUE command were found.
%No {Print,Batch} server <server-name> found	No servers matching the server name you specified in a SHOW SERVER or SET SERVER command were found.
?No qualifiers allowed	The user specified a qualifier on a command that does not allow qualifiers.
%No qualifiers are valid for RESTORE	You attempted to enter a qualifier to the RESTORE mount qualifier. No qualifiers are valid.
?No {read,write,or read/write} access to file <file-spec> [by owner]	The file specifications in a PRINT or SUBMIT command requires access that you or the specified owner does not have.

(continued on next page)

Message **Meaning**

?No room for user on device

In attempting to create a file on a device, or write information to a device, you exceeded some size limit. If it was a nondisk device, this error indicates that the device was full. If it was a disk device, this error indicates one of three conditions:

- The disk as a whole is full.
- You attempted to create a contiguous file, and there is not enough space on the disk. (However, the /CONTIGUOUS qualifier for the commands COPY and CREATE produce only a warning if there is not enough contiguous space.)

If you attempted to create a contiguous file but were unable to, try to create a noncontiguous file of the same size. If you are able to create a noncontiguous file then you can conclude the problem is lack of contiguous space.

- If you have eliminated the first two conditions, then the disk directory has reached its capacity. This capacity is independent of your disk quota. The number varies, depending on the directory cluster size that the system manager assigned when creating the directory and the sizes of the files in it. Large files fill up a directory faster than small files, especially if the files have small cluster sizes. Also, if the directory itself has a large cluster size, it can hold more files and larger files.

?No run-time system

This error can occur if the program you are trying to run requires a run-time system that is not installed.

?Not a valid device

The device name that you gave is invalid for any of the following reasons:

- It is not assigned as a system-wide or user logical name.
- It is not a physical device name of any device that is installed on your system.
- The device name is valid, but not with this command. For example, "INITIALIZE TT:" (you cannot initialize a terminal).

?Not enough available memory

An attempt was made to load a executable program that is too large to run, given the job's private maximum memory size. Either the program must be allowed to expand above a private maximum memory size, or the system manager must increase the job's private memory size maximum to accommodate the program.

?Not enough free memory to create dynamic region

There is not enough memory on the system to create a dynamic region of sufficient size for the operation. If this is a BACKUP, reducing the BLOCKSIZE or BUFFERSIZE parameters, or both, may correct the problem.

?Number not in range <low> or <high>

You typed a number where one is allowed, but the number is not in the valid range. The valid range is from <low> to <high>.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Number too big	You typed a number where one is allowed, but the number is too large. Refer to the command description to find out the largest acceptable value.
?Number too small	You typed a number where one is allowed, but the number is too small. Refer to the command description to find out the smallest acceptable value.
?Pack-id labels don't match	The identification code for the specified disk pack does not match the identification code already on the pack. This message can occur when you try to mount or dismount a disk.
?Page limit exceeds queue's maximum	You specified a /PAGE_LIMIT value in a PRINT command larger than the maximum allowed for the queue.
?Parameter or argument too long	This message occurs if a file specification, text string, or qualifier argument exceeds 255 characters.
%Please answer "YES" or "NO"	Valid responses to the given prompt are YES and NO only.
%Please specify only a device name	You specified something other than a device name when only a device name was valid.
?PPN does not exist	The project-programmer number (PPN) you specified as part of the job specification does not exist.
?PPN needed	The command you typed requires that a PPN be specified.
Previous logical name assignment replaced	An informational message to indicate that an ASSIGN command redefined a previously defined user logical.
{Print,Batch} entry <entry-number><entry-spec> created	Acknowledgment message to PRINT or SUBMIT command.
{Print,Batch} entry <entry-spec> - <error text>	For SET ENTRY, indicates that an entry was not modified for the reason stated in the error text.
?Print/Batch Services already started	You issued the START/QUEUE/MANAGER command; however, the Print/Batch Services (PBS) package is already started.
?Print/Batch services not running	You issued a command that requires action by PBS; however, the package is not running.
{Print,Batch} queue <queue-name> assigned to server <server-name>	Acknowledgment message for ASSIGN/QUEUE command.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
{Print,Batch} queue <queue-name> deassigned from server <server-name>	Acknowledgment message for DEASSIGN/QUEUE command.
{Print,Batch} queue <queue-name> closed	Acknowledgment message for CLOSE/QUEUE command.
{Print,Batch} queue <queue-name> deleted	Acknowledgment message for DELETE/QUEUE command.
%{Print,Batch} queue <queue-name> - <error text>	For SET QUEUE, indicates that a queue was not modified for the reason stated in the error text.
{Print,Batch} queue <queue-name> marked for deletion	Acknowledgment message for DELETE/QUEUE command when the queue you specified still has entries on it. The queue is deleted as soon as it becomes empty.
{Print,Batch} queue <queue-name> opened	Acknowledgment message for OPEN/QUEUE command.
{Print,Batch} queue <queue-name> started	Acknowledgment message for START/QUEUE command.
{Print,Batch} queue <queue-name> stopped	Acknowledgment message for STOP/QUEUE command.
{Print,Batch} server <server-name> {deleted,marked for deletion}	Acknowledgment message for DELETE/SERVER command.
%{Print,Batch} server <server-name> - <error text>	For SET SERVER, indicates that a server was not modified for the reason stated in the error text.
{Print,Batch} server <server-name> initialized [non-]shareable [with forms <form-name>]	Acknowledgment message for INITIALIZE/SERVER command.
{Print,Batch} server <server-name> modified	Acknowledgment message for SET SERVER command.
{Print,Batch} server <server-name> started	Acknowledgment message for START/SERVER command.
{Print,Batch} server <server-name> stopped	Acknowledgment message for STOP/SERVER command.
Print/Batch Services started at <time>	Acknowledgment message when you issue the START/QUEUE/MANAGER command to begin Print/Batch Services.
Print/Batch Services stopped at <time>	Acknowledgment message for STOP/QUEUE/MANAGER command if no jobs are being processed.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message **Meaning**

Print/Batch Services will stop after {completing,aborting} <n> job[s]

Acknowledgment message for STOP/QUEUE/MANAGER command if any jobs are currently in progress.

?Priority exceeds queue's maximum

You specified a priority for a PRINT or SUBMIT entry larger than the maximum allowed for the queue.

?<privilege> privilege required

You typed a command that requires some privilege, and you do not have it.

??Program failure in <program-name>

This message reports a problem in the software. It is followed on the next line by an explanation of the problem. You should verify that the failing program is correctly installed. If necessary, you should then submit an SPR. The SPR should show the dialogue that preceded the message, the exact text of the message, and a list of patches that have been installed in the failing program.

?Program PBS$:PBS.TSK does not exist

The Print/Batch Services (PBS) program was not found when you issued the START/QUEUE/MANAGER command.

?Protection violation

This error can occur for reasons similar to the following:

- You typed a CCL command that your system manager has protected against general use.
- You tried to run a program to which you do not have execute access.
- You tried to read a file to which you do not have read access.
- You tried to write to or delete a file to which you do not have write access.

If this message occurs because of a protection violation with one of your own files, you can use the SET PROTECTION command to change the file's protection code. (However, this error can also occur because of other conditions.)

%Public disk mounted as private

A user with MOUNT privilege mounted a disk initialized as public without specifying /PRIVATE, /PUBLIC, /SHARE, or /NOSHARE. The system mounts the disk as private.

?Qualifier conflicts with file type

You specified the /APPEND qualifier and the file has RMS attributes.

?Qualifier conflicts with parameter

You typed a parameter and a qualifier that should not be present together. For example, with the DEASSIGN command, you specified the /ALL qualifier and a logical name.

?Queue already assigned to server

You issued an ASSIGN/QUEUE for a queue already assigned to the server you specified. Use the SHOW QUEUE/FULL command to see which servers are assigned to a queue.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Queue does not exist	You specified the name of a queue that does not exist.
?Queue is {closed,deleted}	A PRINT or SUBMIT command entry could not be created on a queue because the queue is closed or marked for deletion. Use a different queue.
?Queue is privileged	The queue you specified in a PRINT or SUBMIT command requires one or more privileges that you do not have. Use the SHOW QUEUE/FULL command to display the privileges that the queue requires.
?Queue manager message queue full	There is no room for additional messages to be sent to PBS at this time. Reissue the request.
?Quoted string expected	A quoted string or character was expected and was either missing or was not enclosed in quotes.
%Record too long, line truncated	The record on a READ command was too long. The assignment occurred; however, only the first 255 characters were processed.
?Reserved symbol name	The user attempted to define a local or global symbol which began with a reserved prefix of $, F$ or f$, or delete a reserved global symbol from the symbol table that began with the dollar sign ($).
?Backup set does not contain a valid ANSI label	The Backup set to restore is not a valid ANSI tape.
%Backup set volume label is not <volume-name>	The Backup set volume name does not match the volume name specified by the user.
?Server already exists	You issued the INITIALIZE/SERVER command to define a server that already exists.
?Server does not exist	You specified the name of a print or batch server that does not exist. Use the SHOW SERVER command to list the defined servers.
?Single character expected	A single character inside quotes was expected and more than one character was supplied (for example, SET DATA/END_OF_DATA = "$$").
?Some file data in file <filename> may be missing	The Backup set did not contain the number of file blocks which it was supposed to contain. If there were other errors, then this indicates that the file is missing data. If there are no other errors, submit a SPR with a copy of the Backup set.

(continued on next page)

Table D-1 (Cont.): General RSTS/E Messages

Message	Meaning
?Source device must be the same	Two different source devices were specified for the operation.
?Speed is not allowed for <terminal>	The speed specified is not one of the speeds allowed for this terminal in the TERSPD.SYS file.
??Stack overflow	This message indicates a system problem. The system manager should send in an SPR, giving the dialogue that preceded the message, the text of the message, and a list of patches that have been installed.
?String too long	While in an expression, a string became too long to fit in the 255 available bytes.
?Substitution too complex	The user's request for apostrophe substitution was rejected because the substitution was too complex (it reached the maximum number of substitution iterations allowed on one command).
?Summary record not first record of Backup set	This problem indicates that the Backup set was not written correctly. Submit a SPR including a copy of the Backup set.
?Symbol name conflicts with <symbol-name>	The user specified an assignment whose symbol name or abbreviation point conflicted with an existing symbol definition in the same symbol table.
?Symbol name too long	The symbol name specified by the user exceeded 255 characters.
??Symbol table full	You attempted to define a label or symbol when the local or global symbol table was already full. Note that if you are in a command procedure when your symbol table gets full, the command procedure aborts.
?Syntax error	The command has improper syntax. This occurs when there is not a more specific message describing the syntax error.
?THEN clause required	The action clause was missing on the IF <expression> THEN command or on the ON <severity-level> THEN command.
?THEN keyword required	The THEN keyword was missing on the IF <expression> THEN command or on the ON <severity-level> THEN command.
?This is not the correct output volume	The volume ID of the mounted output volume does not match the volume name specified in the /OUT_LABEL qualifier.
?Time limit exceeds queue's maximum	You specified a /TIME_LIMIT value in a SUBMIT command larger than the maximum allowed for the queue.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
?Too many arguments	You used the notation /qualifier=(arg,arg,...) with a qualifier that accepts only a single argument.
?Too many elements in list	In a list of file specifications or other items (separated by commas or plus signs), you indicated more file specifications than are allowed. For example, you exceeded one of the following limits: • The DIBOL, RENAME, DELETE, and SET PROTECTION commands allow six file specifications. • The COPY command allows six input file specifications and one output file specification. • The PRINT and SUBMIT commands allow up to 11 file specifications.
?Too many files or parameters	You specified too many file specifications in a PRINT or SUBMIT command, or a parameter string in a SUBMIT command is too long to be handled by PBS. Reduce the number of file specifications or use a shorter parameter string.
?Too many logical names assigned	With the ASSIGN command, you exceeded the maximum number of logical names. You can only assign up to four logical names (only three logical names if any of the logical assignments includes a PPN).
?Too many open files on unit	You specified the same magnetic tape or DECtape drive both as input and output files on COPY or APPEND.
?Too many parameters	This message occurs if you specify more command parameters than the command can accept. For example, you specified more than eight parameters with the /PARAMETERS qualifier.
?Too many printers initialized	The device you specified could not be initialized because the maximum number of spooling devices has already been initialized.
?Unable to copy tape command file to disk	The user specified an indirect command file on magnetic tape, and some error occurred when attempting to copy it to a temporary disk file for processing.
??Unable to create DCL work file	DCL's work file for storing the user's symbol tables could not be created on disk. The user may be over quota. Corrective action should be taken before the user continues processing.
??Unable to read DCL work file	The user's symbol tables could not be read in from the DCL work file. If RSTS/E is processing an indirect command file, then control immediately returns to the interactive level.

(continued on next page)

Table D-1 (Cont.): General RSTS/E Messages

Message	Meaning
?Unable to recover part of Backup set	This message indicates that BACKUP was unable to recover part of the Backup set, after trying to use error recovery data. There will also be other messages indicating what data (if any) was detected to be missing.
?Unable to start Print/Batch Services **<error message>**	Some external condition, as the error message on the second line describes, prevented PBS from being started.
??Unable to write DCL work file	The user's symbol tables could not be written out to the DCL work file. The user may be over quota. Corrective action should be taken before the user continues processing. If RSTS/E is processing an indirect command file, then control immediately returns to the interactive level.
?Unbalanced parentheses	Parentheses do not match up (for example, A=((5)+6).
?Undefined label <label>	The label specified on the GOTO command does not exist in the indirect command procedure being executed.
?Undefined symbol	The symbol name specified is not defined in the global or local symbol tables.
?Unexpected character	A character was encountered in a qualifier or a parameter that was not expected. For example, in SET TERMINAL/WIDTH=80FOO, the "F" of "FOO" would be an unexpected character.
?Unexpected error verifying file data for file <filename>	The RSTS/E monitor reported an error in response to BACKUP's request to verify data. Another message describing the problem follows this message.
?Unexpected error writing to output file <filename>	The RSTS/E monitor reported an error when writing to the specified file. Another message describing the problem follows this message.
?Unit number required	This message occurs when you specify a device-name without a device-number (for example, DM: instead of DM0:).
?Unsupported record structure level - record skipped	This message indicates that the version number of a record was higher than the highest version known to BACKUP. If this error occurs trying to restore a Backup set written by a newer version of BACKUP on an older version of BACKUP, the operation may not work properly. If the version of BACKUP writing the tape is older or the same as the one doing the RESTORE, submit an SPR with a copy of the Backup set that generated the error.
?User macros nested too deep	There were more than 50 nested user macros or a user macro called itself.

(continued on next page)

Table D–1 (Cont.): General RSTS/E Messages

Message	Meaning
%Valid qualifiers are /INITIALIZE and /NOINITIALIZE	You entered a qualifier other than /INITIALIZE and /NOINITIALIZE and those were the only valid qualifiers.
?Verification failure for directory information for file <filename>	There was a difference between the directory information (file size, file flags, etc) stored on the Backup set and in the directory structure.
?Verification failure for file attributes for file <filename>	There was a difference between the file attributes stored on the Backup set and in the directory structure.
?Verification failure for file data for file <filename>	The file data stored on the Backup set is different from the file data stored on disk.
%Volume has not been initialized	The initialization request failed, so the output device is not in an initialized state. BACKUP reprompts for an input device.
?Wildcard entry name not allowed	The command you specified does not allow wildcard entry names. Specify the name of a single entry instead.
?Wildcards not allowed	You included a wildcard in a file specification, where wildcards are not allowed.
?Wildcard PPN not allowed	The command you specified does not allow wildcard PPNs. Specify the name of a single PPN instead.
?Wildcard queue name not allowed	The command you specified does not allow wildcard queue names. Specify the name of a single queue instead.
%XOR block missing from Backup set	If this error occurs without any other messages, submit an SPR with a copy of the Backup set. If there are other messages, this has probably occurred because a block of the Backup set was unreadable.
%XOR verification failed - group may be unreliable	This message indicates that BACKUP, during a verify operation, concluded that the recovery data for a group of data blocks does not match the data found in the blocks. If this was not accompanied by other messages, submit a SPR. It it was, consider the data in the Backup set to be unreliable.

Appendix E

SAVRES System Program

The RSTS/E SAVRES system program is a disk backup and copy utility that has operational functions:

- RESTORE recreates a disk from an archived SAVRES tape or disk.
- IMAGE makes a copy of a disk.
- IDENTIFY extracts label information from a SAVRES volume or RSTS/E disk.

E.1 When to Use SAVRES

SAVRES provides a nonselective, fast-volume image capability that requires few operator responses during the operational dialogue. Unlike BACKUP, SAVRES processes entire volumes only and does not allow selective file transfers. When SAVRES finds bad blocks, it requires no operator intervention. For these reasons, use the SAVRES program when you need to:

- Create a fast, reliable image copy of an entire RSTS/E disk
- Restore a previously archived SAVRES set.

E.2 Definitions of SAVRES Terms

To understand this appendix, you should know these terms:

- SAVE format: The format of the output written by a previous SAVE operation and read by a RESTORE operation. These could only be created with versions prior to V9.7.
- SAVE Set: The set of magnetic tapes or disks created by a previous SAVE operation. A SAVE Set must be composed entirely of disks or tapes; it cannot be a combination of the two device types. However, you can mix different drive types within one set, for example, two RK05's and one RK06.
- SAVE volume: One of the magnetic tapes or disks of a SAVE Set.

- SAVE Set Name: One- to six-alphanumeric characters used to identify a SAVE set. By default, the SAVE set name is the same as the Pack ID from which it was created. However, you can specify another name for the SAVE Set.
- LIKE Disks: Like disks are units of the same device size (that is, the same number of data blocks). SAVRES considers the following devices alike:
  - Two RP06 disks
  - An RM02 and an RM03
  - An RP04 and an RP05
  - An RK05 (RK05J) and one unit of an RK05F

  Therefore, if you copied an RM02 to a SAVE Set, the SAVE Set can later be restored either to an RM03 or another RM02.

E.3 Running SAVRES

You use the SAVRES program off line while running the system initialization code INIT.SYS. In reply to the INIT OPTION prompt, type SAVRES. SAVRES responds with the SAV/RES FUNCTION prompt. For example:

```
Option: SAVRES

DD-MMM-YY? 14-MAY-85
HH:MM? 12:15

SAV/RES Function:
```

NOTE

> You must run SAVRES off line to copy (IMAGE) the system disk with which you are currently running on line.

You can also use SAVRES on line, during timesharing. When you do, you must not logically mount the RSTS/E disk(s) on which SAVRES operates. This ensures the integrity of the data on the disk(s). To run SAVRES on line, type RUN $SAVRES. SAVRES then prints its program prompt:

```
$ RUN $SAVRES<RET>
SAV/RES Function:
```

After the SAV/RES FUNCTION prompt appears, as a result of running SAVRES either off line or on line, you are under the control of SAVRES and are ready for restore processing.

Next, select from Table E–1 one of the three operational functions:

- RE[STORE]
- IM[AGE]
- ID[ENTIFY]

Unless you need to exit SAVRES (by pressing Line Feed or Return, or entering Ctrl/Z) or need further help (by typing HE[LP] or pressing the Return key), type one of the three functions on your terminal in response to the SAV/RES FUNCTION prompt. For example:

```
SAV/RES Function: RESTORE<RET>
```

This starts the SAVRES program, which immediately prints the first of several dialogue questions. The section "SAVRES Dialogue" contains sample dialogues for each of the operational functions.

Table E–1 lists the possible responses to the SAVRES prompt.

Table E–1: SAVRES Functions

Function	Description
RE[STORE]	Recreates a RSTS/E file-structured disk from a SAVE Set.
IM[AGE]	Copies a RSTS/E file-structured disk to a like disk.
ID[ENTIFY]	Prints label information and other volume characteristics of a SAVE volume or a RSTS/E file-structured disk.
HE[LP]	Prints a table of SAVRES functions.
<RET> or <LF>	Exits the SAVRES program. If you run SAVRES with a RUN command, pressing Return or Line Feed returns you to your keyboard monitor. If you are running SAVRES off line with INIT.SYS, pressing Return or Line Feed returns you to the INIT OPTION prompt.
?	Displays the "Type HELP for help" message on your terminal.
Ctrl/C	Produces the same result as pressing the Line Feed key.
Ctrl/Z	Produces the same result as pressing the Line Feed key.
other	Produces the same result as pressing the Return key.

Your response to the SAV/RES FUNCTION prompt determines the type of dialogue SAVRES performs. You can respond to the prompt in several ways:

- With the function name, such as IMAGE or RESTORE, or by pressing a key such as Return, as shown in Table E–1. If you enter only a function, the full dialogue follows. See the section "SAVRES Dialogue."
- With the function name and various switches. For example:

  ```
  SAV/RES Function: RESTORE/VERIFY:10-MAY-86
  ```

  If you include switches with the function, the dialogue questions follow only for the devices to be used and the unspecified switches. See the following section.
- With a full function command line. For example:

  ```
  SAV/RES Function: IMAGE DM1:=DM0:MYPAK/VER/NOERR
  ```

  If you enter a complete command line, SAVRES does not ask any dialogue questions. See the section "Full Function Command Line."

SAVRES prints messages on your terminal to tell you its status. Some messages are for information only; other messages are error conditions that may require your attention during the dialogue, mount, transfer, or verification phases of SAVRES.

When the program finds an error during a dialogue or mount operation, it asks you to correct the problem and to retype the appropriate command. Such errors never affect the integrity of the data being transferred.

However, errors that occur while SAVRES is transferring or verifying data can jeopardize data. In this case, SAVRES resolves the problem, aborts the run, or asks you whether to continue or to abort the program. The program gives you information to help you protect your data by printing informational messages and error messages on your terminal.

Finally, SAVRES prints an optional summary report about RESTORE or IMAGE operations at the end of the run. This report describes:

- The operation performed
- Device information
- Start date and time statistics
- Run total statistics, which includes the number of errors found during the operation.

E.4 SAVRES Switches

You can include SAVRES switches in a function response or in a full function command line. There are two types of switches:

- The verification switch corresponds to questions in the SAVRES dialogue and has default settings. If you do not include the switch in a function response, the dialogue asks you for that information.

  If you include one of the switches in a function response, SAVRES does not ask the corresponding question but does ask the question for the switch you did not specify. For example, if you include the /EXPIRATION switch but not /VERIFY, SAVRES does not prompt you for an expiration date but does ask the verification question.

  If you do not include the switches in the full function command line, SAVRES assumes the default settings. For example, if you do not include the /VERIFY switch, SAVRES assumes /NOVERIFY. This means if you do not explicitly include the /VERIFY switch in a full function command line, SAVRES does not compare the data transfer.

- The /STATS and /ERROR switches have no corresponding dialogue questions but do have default settings. Therefore, if you do not specify these switches in a function response or command line, SAVRES assumes the default settings.

Table E–2 lists the allowable SAVRES switches

Table E–2: SAVRES Switches

Switch	Default and Description
VE[RIFY] NOVE[RIFY]	NOVERIFY You can use this switch only for a RESTORE or IMAGE operation. If you specify the /VE[RIFY] switch, SAVRES compares the volume(s) to make sure that the transfer was accurate.
ST[ATS] NOST[ATS]	STATS SAVRES does not automatically print a summary report at the end of a run, unless you specify the /ST[ATS] switch with either a function response or a full function command line. See the section "SAVRES Summary Report."
ER[ROR] NOER[ROR]	ERROR If you do not include the /ERROR switch and a nonfatal error occurs during a SAVRES run, SAVRES prints an error message an continues processing. This is the default condition. You must specify explicitly the /NOERROR switch if you want SAVRES to print an error message and terminate as a result of an of the errors described in Table E-12.

E.5 SAVRES Dialogue

The SAVRES dialogue begins after you run the SAVRES program and respond t the SAV/RES FUNCTION prompt. If you type RESTORE, IMAGE, or IDENTIF the program asks a set of dialogue questions. To give the default answers to thes questions (printed in angle brackets), just press Return. If you press Return in response to a question that has no default, SAVRES repeats the question.

There are two valid responses to a SAVRES program that can change its operating status:

- Ctrl/Z
- Ctrl/C

If you enter Ctrl/Z at any point in the dialogue, SAVRES returns to the previous question. A Ctrl/C response aborts the run immediately and returns you to the INIT OPTION prompt if you are running off line under the control of the INIT.SYS program, or to your keyboard monitor prompt if you are running SAVRES on line. SAVRES automatically returns to the SAV/RES FUNCTION prompt after the execution of a specified function when you are operating either on line or off line. Exit from the INIT version of SAVRES to the OPTION promp by pressing Line Feed or Return. Type Ctrl/Z to the SAV/RES FUNCTION prompt to exit the online version of SAVRES.

E.5.1 Device Specifications

SAVRES requires you to use device specifications as responses to several dialogue questions. The general format is:

<device>[:[<id>]][<switch(es)>]

Device is a two-character device type (for example DU) followed by a one- or two-digit unit number in the range 0 to 15. The identification can be either a SAVE Set Name or a disk pack identification; the switches are any of those from Table E–3.

Table E–3: SAVRES Device Specification Switches

Switch	Description
/SCR[ATCH]	The /SCRATCH switch is legal only on an output volume device specification. If you use the /SCRATCH switch, SAVRES bypasses most volume label checking. The program always checks the label to see if an output volume contains a SAVE Set written at the current density It checks destination disks for bad block information.
/DEN[SITY]:[800] [1600] [6250]	The /DENSITY switch is legal only for magnetic tape. With this switch, you can specify the density at which the tape is to be read. If SAVRES cannot read the tape at the specified density, it tries to read the tape at the other legal density setting.

E.5.2 Privileges Required

You need RDNFS, WRTNFS, and SYSMOD privileges to gain read/write non-file-structured access to a disk. If a device is restricted, you also need DEVICE privilege to gain access to that device.

E.5.3 Checking the Input Volume

SAVRES checks the input volume before transferring any actual data and warns you of any problem it foresees in restoring or copying the volume. All of the input volume error messages that are printed at this point are warning messages only; you can recover from them by following appropriate procedures.

Table E–4 lists and describes the error messages that result from a problem witl an input volume.

Table E–4: SAVRES Input Volume Error Messages

Message and Meaning

%%% **Input disk has only nn% free clusters.**
Mount it anyway <NO>?

The input disk has very few clusters that are not allocated. There must be at least as many good pack clusters on an output RSTS/E disk as there are clusters to be transferred from the original input disk. This message indicates that you may encounter problems copying the disk. Type NO or press RETURN to return to the previous device prompt. If you type YES, SAVRES proceeds with the IMAGE operation. The program also checks th output volume to see if the transfer can be made (see the section "Checkin the Output Volume"). At this point, you may want to free up space on the input disk by deleting unnecessary files. This increases your chances of completing a successful IMAGE.

%%% **This is not the correct volume.**

The pack ID or SAVE Set Name you included with the input device specification does not match the name already written on the volume. If SAVRE finds that the names do not match, it prints the warning message and then issues the input device prompt again. A dismount request follows the warn ing message and lists useful information from the volume label of a SAVE Set or a RSTS/E file-structured disk.

%%% **This volume has no label.**

The input volume is not a RSTS/E disk or a SAVE Set. SAVRES prints the input device prompt again after issuing the warning message.

%%% **Input disk should be rebuilt.**

You removed a disk from a drive without logically dismounting it and then attempted to remount it. Perform the clean operation with the ONLCLN program or the REFRESH option of INIT.SYS. See the *RSTS/E System Installation and Update Guide* for information on REFRESH. See the *RSTS/E Utilities Reference Manual* for information on ONLCLN.

E.5.4 Checking the Output Volume

SAVRES checks the output volume before restoring or copying a RSTS/E disk to make sure that each operation ends with the least chance of error. After checking the output volume, SAVRES displays messages about the expiration date, the label, and the available data space on the destination device. Some of the messages are only for your information; the rest warn you of serious output device problems.

When the output volume you are using is labeled a RSTS/E disk, SAVRES prints the message:

```
*** The volume on dev: is a RSTS disk
*** Pack ID is <xxxxxx>
*** Pack will be reinitialized
Mount it anyway <NO>?
```

If you type NO or press Line Feed or Return, SAVRES returns to the previous output prompt. A YES response indicates you want to use the present volume.

If you restore (RESTORE) or copy (IMAGE) a disk, SAVRES checks whether the total number of clusters on the output disk minus the known bad blocks is greater than or equal to the number of allocated clusters on the original source disk. If there is not enough space and the transfer cannot be made, it prints the following message and then returns to the previous device prompt:

```
%%% Too many bad blocks on output disk.
```

However, if the program decides there are very few free blocks, it prints:

```
%%% Only nn% of the output disk clusters are available for relocation.
Mount it anyway <NO>?
```

You must decide whether the number of free clusters is large enough to permit a successful SAVRES operation. If you decide it is not, type NO or press Line Feed or Return to return to the output prompt. Type YES if you want SAVRES to continue the operation.

NOTE

> Whenever you get this warning message, DIGITAL recommends you use another disk for output. If you do not, there is a greater than normal chance the operation will not succeed.

SAVRES normally transfers each allocated pack cluster from the original source disk to the same cluster on the output RSTS/E disk. If SAVRES finds the output cluster is bad, it tries to relocate the data. If the cluster size of the item being moved is larger than the pack cluster size, there must be enough contiguous free pack clusters on the output disk to accommodate the entire source cluster. If there is not, SAVRES aborts the operation.

If the specified output is not a RSTS/E disk, the program prints the new SAVE Set Name and proceeds without further notice. It does not check the output volume label if you specified the /SCRATCH switch. It always attempts to recover bad block information from a RSTS/E file-structured and SAVE disk.

E.5.5 Restoring a RSTS/E Disk using the RESTORE Dialogue

To recreate a RSTS/E file-structured disk from a SAVE Set, you must perform the RESTORE operation. Because SAVRES writes the SAVE Set in the SAVE format, no other RSTS/E software can perform this rebuild process. Before transferring any data, SAVRES tries to extract bad block information from the destination disk and, if it finds any bad blocks, incorporates them into the new BADB.SYS file.

The SAVRES program may find bad blocks during a RESTORE run. If it does, it scans the SATT.SYS file to determine if there is a place where the block corresponding to a bad block can be relocated. If a large enough area exists on the destination disk, the program simply moves the data to the new location and modifies the directory information accordingly.

The relocation of blocks may require RESTORE to change a file's characteristics. For example, this can occur if the program finds a bad cluster while transferring a contiguous file. To relocate the cluster that falls on the bad block, SAVRES must make the file noncontiguous. Changes of this type can cause problems, but RESTORE preserves the data and informs you of these adjustments.

The operation ends if there is no way RESTORE can relocate the data cluster. This can occur if:

- No unallocated pack cluster remains on the disk
- The file cluster size is larger than the pack cluster size and there is not an equivalent number of contiguous pack clusters left on the disk

Normally, SAVRES uses the name of the input SAVE Set as the Pack ID of the output volume. After you answer the From Device? question, SAVRES prints a message showing you the name it plans to use. The general format is:

*** SAVE Set name/Default Pack ID is xxxxxx

After you successfully respond to the TO RSTS DISK question, SAVRES prints the message:

```
*** Pack ID is xxxxxx
```

Table E–5 lists the RESTORE dialogue questions. An example of a RESTORE procedure follows the dialogue description.

Table E–5: RESTORE Dialogue Questions

Question	Response and Description
From Device?	<devspec> Data is transferred from the specified device. If you include a SAVE Set Name in the device specification (see section "Device Specifications"), SAVE/RESTORE checks whether it matches the name on the device. If it does not, the program prints a warning message (see section "Checking the Input Volume") and prints the From Device? question again. Type 2 characters to specify the input device type, followed by the unit number. Include a SAVE Set Name if you want SAVE/RESTORE to make sure it matches the one on the volume. The device must be physically mounted and write-locked.
To RSTS Dx: Disk?	<devspec> Data from the previously specified SAVE Set is restored to this disk. If you include a Pack ID in the device specification, it becomes the Pack ID of the restored disk. Otherwise, SAVE/RESTORE uses the SAVE Set Name (which may also be the Pack ID of the original source disk). Physically mount and write-enable the disk. Type 2 characters to specify the output device type, followed by the unit number. Include a Pack ID if you want to override the default. This drive should have a SCRATCH volume mounted, and must be write-enabled.
Verify(Yes or No)<NO>?	Y[ES] or N[O] This is the dialogue form of the /VERIFY switch. If you type YES, SAVE/RESTORE compares the output with the input at the end of each input volume to make sure the volumes are equivalent. Type YES if you want SAVRES to compare the input and output volumes at the end of each input volume. Type NO or press Line Feed or Return if you do not want SAVRES to perform this verification.
Proceed(Yes or No)?	Y[ES] or N[O] This question lets you either double check your dialogue responses and abort the operation if you have made errors, or proceed with the RESTORE operation. Type YES to proceed with the operation. Type NO to abort and return to the SAV/RES Function: prompt.

NOTE

If you are using the output of a RESTORE operation as a system disk, you must use the INIT.SYS INSTALL option to reinstall the desired monitor Save Image Library (SIL). You must also reenter any necessary HARDWR suboptions (such as changing Hertz) because the INSTALL option clears various items set up by the HARDWR suboptions. This procedure is necessary because a disk can be saved with SAVE from one system disk and restored with RESTORE on a system with a different hardware configuration.

The following RESTORE example shows the restoration of an RM03 disk from the tape SAVE Set created in the previous example. The letters are keyed to the explanation that follows this sample.

The RESTORE steps are:

1. Mount the two SAVE Set volumes on drive MM1: and MM2:, write-protected.
2. Mount the destination disk on DR1:, write-enabled.
3. Type RUN $SAVRES and press Return.

```
❶  $ RUN $SAVRES<RET>
❷  SAV/RES Function: RESTORE<RET>
❸  From device? MM1:<RET>
   *** SAVE Set Name/default Pack ID is SAVSET
❹  To RSTS DR: Disk? DR1:TSTPAK<RET>
   *** The volume on DR1: is a RSTS disk
   *** Pack ID is MYPAK
   *** Pack will be reinitialized
   Mount it anyway <NO>? Y<RET>
   *** Pack ID is TSTPAK
❺  Verify (Yes or No) <NO>? <RET>
❻  Proceed (Yes or No)? Y<RET>
   *** Begin RESTORE from MM1: to DR1: at 08:28 PM
    Dismount Device:       MM1:
           Set Name:       SAVSET
              Seq #:       1
            Density:       800
      Creation date:       Friday, 10-May-85
    Expiration date:       Saturday, 10-May-86
   Mount volume # 2 of SAVE Set SAVSET
   Device ? MM2:<RET>
   Proceed (Yes or No)? Y<RET>
   *** Begin RESTORE from MM2: to DR1: at 08:32 PM
    Dismount Device:       MM2:
           Set Name:       SAVSET
              Seq #:       2
            Density:       800
      Creation date:       Friday, 10-May-85
    Expiration date:       Saturday, 10-May-86
    Dismount Device:       DR1:
            Pack ID:       TSTPAK
   Pack Clustersize:       8
      Creation date:       Monday, 15-Apr-85
           Please label this volume!
   --- RESTORE operation completed at 08:37 PM
```

```
❼              Summary Report
   RESTORE of SAVE Set SAVSET to DR1:TSTPAK

       Input Device:       Magtape
           Set Name:       SAVSET
       # of volumes:       2
            Density:       800
      Creation date:       Friday, 10-May-85
    Expiration date:       Saturday, 10-May-86

      Output Device:       DR1:
            Pack ID:       TSTPAK
   Pack Clustersize:       8
      Creation date:       Monday, 15-Apr-85

   RESTORE started on Friday, 10-May-85, at 08:28 PM
❽             Run Statistics

   Transfer Totals:

        Total of 54184 blocks transferred

   Error Totals:

        Total of     0 new bad blocks encountered on destination.

        0 files structurally altered.

   Timing Totals:

        Total elapsed time:   0 hrs.,    9 mins.,   36 secs.

        Total wait    time:   0 hrs.,    0 mins.,   16 secs.

        Total process time:   0 hrs.,    9 mins.,   20 secs.

❾  SAV/RES Function: <Ctrl/Z>
   $
```

The explanation of the RESTORE example follows:

❶ Type RUN $SAVRES, and then press Return to run the SAVRES program.

❷ Begin the RESTORE operation by typing RESTORE and pressing Return in response to the initial SAVRES prompt. Because you need to recreate an RM03 from a tape SAVE Set, you select the RESTORE function.

❸ The RESTORE dialogue begins when it prints the From Device? question. In response, you specify the device name and the unit number of the SAVE Set medium and press Return. Because the input medium is tape, mounted on drive number 1, you enter MM1:. If you had included a SAVE Set Name in the device specification, RESTORE would have checked it against the name already on the set. If they had not matched, RESTORE would have issued a warning message informing you of the discrepancy. After you terminate your response by pressing Return, RESTORE reads the SAVE Set Name from the input medium and displays it on your terminal.

❹ Because in this RESTORE operation you are recreating an RM03 disk, you respond to the To RSTS DR: Disk? question by typing DR1:TSTPAK. By your response, you indicate that the device mnemonic of the RM03 is DR, that it is mounted on drive number 1, and that you want it to have a Pack ID of TSTPAK. After pressing Return, RESTORE checks the Pack ID of the output disk. It then prints a message showing you that the disk is a RSTS/E disk, has a Pack ID of MYPAK, and that RESTORE will reinitialize it. RESTORE issues a Mount It Anyway? question that gives you the option to mount the disk or not. Because the data on the disk is no longer valuable, answer Y and press Return. RESTORE then initializes the disk with the new Pack ID.

❺ The comparison of the input data to the data already transferred to the destination disk is not essential during this particular run; therefore, you press Return to accept the No default. When you ask RESTORE to verify the transferred data, you are essentially requesting that RESTORE perform two operations rather than one. That is, you make the program perform two passes on the data, once for the transfer and once for the comparison.

❻ No mistakes were made during the dialogue phase; therefore, RESTORE proceeds with the operation in reply to your Y response. After you press Return, RESTORE displays a message to indicate it has begun the transfer. After RESTORE completes the first volume, it prints input medium information, asks you to mount the next volume, and asks if you want to continue the operation. Again, because there are no errors in the dialogue, RESTORE proceeds. RESTORE moves the remaining data to the second volume, issues the input and output dismount messages, and informs you that the operation is completed.

❼ The Summary Report prints the SAVE Set Name, the Pack ID, and label information for both the input and output medium, in addition to the date and time of the RESTORE operation.

❽ The Run Statistics Report prints the number of blocks transferred, the number of bad blocks encountered, and the times required to perform various operations.

❾ RESTORE returns to the SAV/RES FUNCTION prompt when the operation has ended. Enter Ctrl/Z to return to your default keyboard monitor (DCL in this example).

E.5.6 Copying a RSTS/E Disk using the IMAGE Dialogue

The IMAGE function lets you make an equivalent copy of a RSTS/E file-structured disk; however, only disk-to-LIKE-disk transfers are legal when using the IMAGE operation. LIKE disks are units that have exactly the same device size. The SAVRES program handles bad block recovery during an IMAGE operation in the same way as RESTORE.

Table E–6 lists the IMAGE dialogue questions. An example of the IMAGE operation follows the dialogue description.

NOTE

> If you are using the output of an IMAGE operation as a system disk, you must use the INIT.SYS INSTALL option to reinstall the desired monitor Save Image Library (SIL). You must also reenter any necessary HARDWR suboptions (such as changing Hertz) because the INSTALL option clears various items set up by the HARDWR suboptions. This procedure is necessary because you can make a copy of a disk on one system and use the copied disk on another.

Table E–6: IMAGE Dialogue Questions

Question	Response and Description
From RSTS Disk?	<devspec> SAVRES transfers all data from the disk specified in the device specification response. If you include a Pack ID in the device specification, SAVRES checks to see if it matches the one actually on the disk. If it does not, the program prints a warning message (see section "Checking the Input Volume") and prints the From RSTS Disk? question again. Type 2 characters to specify the input device type, followed by the unit number. Include a Pack ID if you want SAVRES to make sure it matches the one on the volume. This device must be physically mounted and write-locked.
To RSTS Dx: Disk?	<devspec> SAVRES transfers all information from the previously specified source disk to this disk. If you include a Pack ID in the device specification, it becomes the Pack ID of the output disk. Otherwise, SAVRES uses the Pack ID of the source disk. Type 2 characters to specify the output device type, followed by the unit number. Include a Pack ID if you want to override the default. This drive should have a SCRATCH volume mounted, and must be write-enabled. Note that the device mnemonic you specify in response to the From RSTS Disk? question appears in place of the Dx in the To RSTS Dx: Disk? question. In some cases, in your device specification response to the To RSTS Dx: Disk? question, you may specify a device mnemonic other than the one that appeared for Dx. See the discussion of LIKE disks in the section "Definitions of SAVRES Terms."
Verify(Yes or No)<NO>?	Y[es] or N[o] This is the dialogue form of the VER[IFY] switch. If you type YES, SAVRES compares the output with the input at the end of each output volume to ensure that the volumes are equivalent. If you type NO, SAVRES moves on to the next question without performing a verify pass. Type YES if you want SAVRES to compare the input and output volumes at the end of the IMAGE copy operation. Type NO or press Line Feed or Return if you do not want SAVRES to perform the verification.
Proceed(Yes or No)?	Y[ES] or N[O] This question lets you either double check your dialogue responses and abort the operation if you have made any errors or proceed with the IMAGE operation. Type YES to proceed with the operation. Type NO to abort and return to the SAV/RES Function: prompt.

After you answer the From RSTS Disk? question, the program prints a message in the following general format to tell you the Pack ID it intends to use:

```
*** Input Pack ID/default Output Pack ID is XXXXXX
```

If you do not want the output disk to have the same name as the input volume, specify a different Pack ID in the <devspec> response to the To RSTS DX: Disk? question. After you select the Pack ID and answer the question, SAVRES prints:

```
*** Output Pack ID is XXXXXX
```

The following procedure describes an IMAGE operation that copies an RA80 to an RA80 and verifies the transfer in the same pass. The letters are keyed to the explanation that follows this sample.

The IMAGE function steps are:

1. Mount the source disk on drive DU0:, write-locked.
2. Mount the destination disk on drive DU1:, write-enabled.
3. Type RUN $SAVRES and press Return.

```
❶  $ RUN $SAVRES<RET>

❷  SAV/RES Function: IMAGE/NOERROR<RET>

❸  From RSTS disk? DU0:<RET>

   *** Input Pack ID/default Output Pack ID is SOURCE

❹  To RSTS DU: Disk? DU1:DESTIN<RET>

   *** The volume on DU1: is a RSTS disk
   *** Pack ID is OUTPUT

   *** Pack will be reinitialized
   Mount it anyway <NO>? YES<RET>

   *** Output Pack ID is DESTIN

❺  Verify (Yes or No) <NO>? YES<RET>

❻  Proceed (Yes or No)? YES<RET>

   *** Begin IMAGE copy from DU0: to DU1: at 11:02 AM
   *** Begin VERIFY pass from DU0: to DU1: at 11:04 AM
   *** 0 differences found

    Dismount Device:      DU1:
            Pack ID:      DESTIN
   Pack Clustersize:      8
      Creation date:      Friday, 10-May-85

           Please label this volume!

   --- IMAGE copy operation completed at 11:07 AM

❼               Summary Report

   IMAGE copy of DU0:SOURCE to DU1:DESTIN

       Input Device:      DU0:
            Pack ID:      SOURCE
   Pack Clustersize:      8
      Creation date:      Friday, 10-May-85

      Output Device:      DU1:
            Pack ID:      DESTIN
   Pack Clustersize:      8
      Creation date:      Friday, 10-May-85

IMAGE copy started on Friday, 10-May-85, at 11:02 AM


❽             Run Statistics

   Transfer Totals:

        Total of 24400 blocks transferred

   Error Totals:

        Total of     0 bad compares.

        Total of     0 new bad blocks encountered on source.

        Total of     0 new bad blocks encountered on destination.
```

```
          0 files structurally altered.

     Timing Totals:

          Total elapsed time:   0 hrs.,   5 mins.,   22 secs.

          Total wait    time:   0 hrs.,   0 mins.,    0 secs.

          Total process time:   0 hrs.,   5 mins.,   22 secs.

❾  SAV/RES Function: <Ctrl/Z>
```

The explanation of the IMAGE example follows:

❶ Type RUN $SAVRES and press Return to run the SAVRES program.

❷ SAVRES prints the SAV/RES FUNCTION prompt to determine which function you want to perform. To create a copy of an RA80, select the IMAGE function by typing IM/NOERROR and pressing Return. Attach the /NOERROR switch to insure that no data is lost. The /NOERROR switch causes SAVRES to end the operation whenever the program finds a run-time error. Before the program aborts, it prints an error message indicating the source of the problem.

❸ In reply to the From RSTS Disk? question, specify the disk from which you are to extract the data. Enter the device name and unit number of the input disk and end the response by pressing Return. After this response, IMAGE reads the Pack ID from the input disk and displays it on your terminal.

❹ IMAGE requests the device name and unit number of the output disk. Type DB1:DESTIN to indicate that the RA80 is on drive 1 and you want DESTIN to be the new Pack ID. As soon as you press Return, SAVRES tells you that the output medium is a RSTS/E disk, its Pack ID is OUTPUT, and it will be initialized. SAVRES initializes the disk when you type Y and press Return in response to the Mount it anyway? question. A message that SAVRES prints informs you that the disk was initialized to the specified Pack ID.

❺ To insure against losing data, type YES to have IMAGE determine if the input data equals the data that was transferred to the output disk. If IMAGE finds that the data is not equivalent, it informs you of these differences.

❻ You are satisfied that you made no errors in the dialogue sequence and feel that the operation should proceed. After you press Return, IMAGE informs you when it begins to transfer data and when it begins the verify pass. After completing the data comparison, IMAGE prints the number of differences found between the two devices. It then issues the output device dismount message. Immediately after IMAGE prints the dismount request, it prints a message to show you the copy operation ended successfully.

❼ IMAGE prints the Summary Report unless you suppress it by using the /NOSTATS switch. The report includes the Pack ID and label information of each disk, in addition to the date and time that the IMAGE operation was run.

❽ The Run Statistics Report provides a total figure for the number of blocks transferred, the errors encountered, and the times required to perform the IMAGE operation.

❾ SAVRES returns you to the SAV/RES FUNCTION prompt when the operation ends. Enter Ctrl/Z to return to your keyboard monitor (DCL in this example).

E.5.7 IDENTIFY Dialogue

If you want label information from a SAVE Set volume or a RSTS/E file-structured disk, use the IDENTIFY function. SAVRES asks only one question before the IDENTIFY operation begins.

Table E–7 lists the IDENTIFY dialogue question. Examples of the IDENTIFY operation appear after the table.

Table E–7: IDENTIFY Dialogue Question

Question	Response and Description
From Device?	<devspec> SAVRES prints the device label characteristics of the SAVE Set or RSTS/E file-structured disk for the device specified in the dialogue response. If you include a SAVE Set Name or Pack ID, SAVRES checks to see if it matches the name on the device. If it does not, the program prints a warning message (see the section "Checking the Input Volume") and then prints the From Device? question again. Type 2 characters to specify the input device, followed by the unit number. Include a SAVE Set Name or Pack ID if you want SAVRES to make sure it matches the one on the volume. You must mount and write-lock the device (for protection) before responding to the From Device? question.

The following IDENTIFY example shows the label characteristics of both a SAVE Set and a RSTS/E disk. The letters are keyed to the explanation that follows this sample.

The IDENTIFY procedures are:

1. Mount a SAVE Set volume on MM1:, write-locked.
2. Mount the RSTS/E disk on drive DR1:, write-locked.
3. Type RUN $SAVRES and press Return.

```
❶  $ RUN $SAVRES<RET>
❷  SAV/RES Function: ID MM1:<RET>
❸            Device:       MM1:
           Set Name:       SAVSET
              Seq #:       1
            Density:       1600
      Creation date:       Friday, 10-May-85
    Expiration date:       Saturday, 10-May-86
❹  SAV/RES Function: ID DR1:<RET>
             Device:       DR1:
            Pack ID:       TSTPAK
   Pack Clustersize:       8
      Creation date:       Monday, 15-Apr-85
❺  SAV/RES Function: <Ctrl/Z>
   $
```

❶ Type RUN $SAVRES and press Return to run the SAVRES program.

❷ The SAVRES program prints the SAV/RES FUNCTION prompt. You respond to it with the first two letters of the IDENTIFY function and with the device name and unit number of the medium you want to identify. Include a Pack ID if you want SAVRES to compare the ID you specify with the one on the input volume.

❸ IDENTIFY prints label information for the tape SAVE Set.

❹ SAVRES returns to the SAV/RES FUNCTION prompt. In response, you ask SAVRES to identify the disk pack mounted on DR1:. SAVRES prints the information as soon as you end your response.

❺ Enter Ctrl/Z to return to your keyboard monitor (DCL in this example).

E.5.8 Full Function Command Line

Use the full function command line format if you do not want to answer each dialogue question separately. The general format is:

SAV/RES Function:<function><outdevspec>=<indevspec>[<switch(es)>]

Function can be RE[STORE], IM[AGE], or ID[ENTIFY] and the switches can be any of those listed as valid with the associated function (see section "SAVRES Switches"). See the section "Device Specifications" for an explanation of the device specifications.

If you specify a full command in response to the SAV/RES FUNCTION prompt, SAVRES prints only the PROCEED question (except during an IDENTIFY question):

```
Proceed (Yes or No)?
```

You must type either YES or NO to this question. Type YES to have SAVRES begin the operation; type NO to abort.

SAVRES does the same standard volume label checking during the full function command line operation as it does during the longer dialogue procedure. If a discrepancy occurs, SAVRES prints the error message and then returns to the SAV/RES FUNCTION prompt. At that point, you must type the full function command line again.

E.6 SAVRES and Booting

SAVRES always attempts to create bootable media. The output of a RESTORE or IMAGE copy operation is bootable if the original disk contained an INIT.SYS file in [0,1] (even if the original disk was not bootable).

If a SAVE volume is booted, a portion of INIT.SYS is loaded, allowing you to perform the following INIT options:

- SAVRES
- DSKINT
- BOOT
- HARDWR option specifying the LIST suboption

The types of SAVRES operations you can perform depend on the type of device you boot from. You can run all three SAVRES options if you boot from a RSTS/E disk or a RSTS/E magnetic tape or SAVE volume. If you boot a SAVE volume and then perform a RESTORE operation, SAVRES prints a message at completion of the run:

```
*** Please boot from the system disk
Boot device?
```

You must specify the device name and unit number of a bootable device.

Table E–8 summarizes the SAVRES operations you can perform while booting various devices.

Table E–8: Booting RSTS/E and SAVE Set Volumes

Device Booted From	**SAVRES Operations You Can Perform**	**SAVRES Operations You Can Perform Afterwards**
RSTS/E Disk or Magnetic Tape	IDENTIFY, IMAGE, RESTORE	IDENTIFY, IMAGE, RESTORE
SAVE Set Volume (Disk or Tape)	IDENTIFY, IMAGE	IDENTIFY, IMAGE, RESTORE
SAVE Set Volume (Disk or Tape)	RESTORE	None—You receive a message that states you must boot from the system disk.

NOTE

You cannot use the console bootstrap to boot a magnetic tape SAVE Set if:

- The tape has a density of 1600 bpi.
- The drive the tape is mounted on has a TM02 formatter. (Use the HARDWR LIST suboption of INIT to determine the type of formatters on your drives.)

However, you can boot a 1600 bpi tape on a drive with a TM02 formatter if you use the BOOT OPTION prompt of INIT.SYS. That is, you can boot under these circumstances using software but not hardware boot procedures. As an additional restriction, you can boot a magnetic tape SAVE Set only on the following drives: TS11, TU45, TU80, TU16, TU77, and TE16.

E.7 Operator Interface During Processing

SAVRES requires operator intervention at certain points during processing. The operator must mount and dismount devices, reset hung devices, and abort operations when necessary. The following sections describe these procedures.

E.7.1 Mounting and Dismounting Volumes

A SAVE Set can contain more than a single volume. Because you may be processing multivolume SAVE Sets, the SAVRES program contains procedures that allow you to mount the additional volumes in a set. When SAVRES needs an additional volume to complete a RESTORE operation, it prompts you with the message:

```
Mount volume #nn of SAVE Set <setnam>
Device?

   .
   .
   .

Proceed (Yes or No)?
```

The letters nn represent the volume sequence number and <setnam> indicates the SAVE Set Name and unit number of the drive on which the next volume has been readied. If you mount a device of the same type, you can respond to this prompt only with a unit number. After you press Return to terminate your device specification, SAVRES performs volume label checking (see section "Checking the Input Volume"). At this point, if the additional volume you mount violates any SAVRES rule, the program prints the error message immediately after the Device? question.

SAVRES prints a dismount message when:

- SAVRES completes the data transfer to the output volume in an IMAGE copy operation
- You mount an incorrect volume

Note that the dismount message SAVRES generates for a SAVE Set (Format 1) differs from a dismount message printed for a RSTS/E disk (Format 2):

Format 1

```
 Dismount Device:      <device>
        Set Name:      <setnam>
           Seq #:      nn
         Density:      [ 800]
                       [1600]
   Creation Date:      <(day-of-week), dd-mmm-yy>
 Expiration Date:      <(day-of-week), dd-mmm-yy>
```

Format 2

```
 Dismount Device:      <device>
         Pack ID:      <packid>
Pack Clustersize:      mm
   Creation Date:      <(day of week), dd-mmm-yy>
```

The <device> field represents a two character device mnemonic followed by a single digit unit number. The SAVE Set Name replaces <setnam>, and the disk pack identification replaces <packid> during a valid SAVRES operation. Density figures are printed only for magnetic tape SAVE Sets.

E.7.2 Re-Accessing Devices

During an operation you may find that SAVRES does not have access to a previously accessible device. If a device does become hung or is write-protected, the program notifies you in the warning message:

```
%%% Device hung or write-locked
Retry (Yes or No)?
```

Correct the condition causing the problem if possible, and then type YES to attempt to gain access to the device. Type NO to abort the attempt.

E.7.3 Aborting SAVRES

To end a SAVRES operation at any time, enter Ctrl/C. This returns control to you keyboard monitor if you running SAVRES on line. If you are running under the control of the INIT.SYS program and the operation ends, INIT.SYS returns you to its OPTION prompt. An abort during a RESTORE or IMAGE operation leave the output volume unusable. However, you do not jeopardize any IDENTIFY volumes if you choose to abort this operation.

E.8 SAVRES Summary Report

SAVRES prints a Summary Report at the end of RESTORE, and IMAGE operations. The report identifies the operation that was run, lists characteristics of the input and output devices, and generates a list of data transfer, error, and timing statistics. It also documents the time and date of the operation.

E.8.1 Summary Report Format and Example

The format of the Summary Report varies with the operation. At times the Input Device characteristics (see Figure E–1) identify a SAVE Set, as in the case of a RESTORE operation.' Thus, depending on the operation, the SAVE Set format may be represented under either the Input or Output Device category. The RSTS/E disk format may be printed also under the Input or Output Device categories. Figure E–1 shows the general format of the SAVRES Summary Report.

Figure E–1: Summary Report Format

```
IMAGE       of    disk<packid>         to       disk<packid>
RESTORE           SAVE Set <setnam>             SAVE Set<setname>

    Input Device:       <device>
        Pack ID:        <packid>
Pack Clustersize:       nn                     *RSTS/E disk format
   Creation date:       <day of week, dd-mmm-yy>

                        [magnetic tape]
   Output Device:       [disk]
        Set Name:       <setnam>               *SAVE Set format
    # of volumes:       nn
         Density:       [ 800]
                        [1600]
   Creation date:       <day of week, dd-mmm-yy>
 Expiration date:       <day of week, dd-mmm-yy>

            Run Statistics

Transfer Totals:

     Total of nnnnnnn blocks transferred

Error Totals:

     Total of nnn bad compares.

     Total of nnn new bad blocks encountered on source.

     Total of nnn new bad blocks on destination disk.

     Total of nnn files structurally altered.

Timing Totals:

     Total elapsed time:   nn hrs.,  nn mins.,  nn secs.
     Total wait    time:   nn hrs.,  nn mins.,  nn secs.
     Total process time:   nn hrs.,  nn mins.,  nn secs.
```

E.8.2 Summary Report Run Statistics

Table E–9 lists and describes the run statistics that appear at the end of a RESTORE or IMAGE operation. Because each SAVRES operation prints its own set of run statistics, Table E–9 also identifies the operations in which each run statistic is printed. The operations are identified according to the following conventions:

R = RESTORE operation

I = IMAGE operation

Table E–9: Summary Report Run Totals

Type of Statistic	Operation and Description
Transfer Statistics	
Total of blocks transferred	This number refers to the total number of blocks transferred (directory blocks, file blocks). In a SAVE operation an extra set of directory blocks is stored at the end of a SAVE Set. The total number of blocks transferred includes these extra blocks. [RI]
Error Statistics	
Total of bad compares	The total number of errors that occurred during the VERIFY phase. [RI]
Total of new bad block errors	The number of previously unreported bad blocks encountered on the source disk. [RI]
Total of bad block errors on destination disk	The number of previously unreported bad blocks encountered on the destination disk. [RI]
Total of files structurally altered	The number of files that were structurally altered to enable relocation on the destination disk. (This means contiguous files were made noncontiguous or placed files were moved.) [RI]
Time Statistics	
Total elapsed time (TE)	The total amount of real clock time that elapsed during the run. [RI]
Total wait time (TW)	The amount of real clock time that elapsed while waiting for operator responses and magnetic tape rewinds. [RI]
Total process time (TP)	The amount of real clock time that was used for processing (TE-TW=TP). [RI]

E.9 SAVRES Error Handling

SAVRES issues dialogue, mount, transfer, and verification error messages. When you find an error message during a dialogue or a mount, SAVRES asks you to correct the condition, where appropriate, and then asks you to reenter the command. Errors you encounter during either of these phases never affect the integrity of the data being transferred.

NOTE

> When SAVRES reports an error during a transfer or verification phase, it has detected a problem that might corrupt the data being transferred. Depending on the nature of the error, a single file, or perhaps an entire volume, might be affected. Generally, you encounter fewer critical errors during a dialogue or mount phase than in a transfer or verification operation.

SAVRES does not report an error when it finds a bad block on the output device, unless it is unable to reallocate that block in a way that is transparent to you. For instance, the one-block file SHIFT.LES is moved without notice if a free cluster can be found. On the other hand, the program issues an error message if it must move the placed file, MOVEME.NOT.

The /NOERROR switch causes SAVRES to abort the run if it finds any condition that warrants an error message. If you do not specify the /NOERROR switch, the program takes whatever action necessary. SAVRES always informs you of any corrective action taken so you have the option of aborting the operation.

The most important problem that SAVRES encounters during a RESTORE or IMAGE copy operation is finding a bad block on a RSTS/E file-structured disk when it wants to write to that block. The program attempts to correct the problem by performing the following actions in the order listed:

1. Marks a contiguous file as noncontiguous and relocates the bad cluster but not the entire file.
2. Marks the file as nonplaced if the bad cluster occurs at the beginning of a placed file.
3. Finds enough contiguous free pack clusters on the output disk to hold the number of clusters that must be moved. In other words, if the cluster size of a file is 16, SAVRES finds 16 free contiguous blocks for the cluster. It proceeds if the reallocation worked and aborts if it has not worked.

SAVRES always tells you about the changes it makes to a file that are caused by this bad block processing. If there is no way to relocate a cluster, the program aborts the operation. Unless you specify the /NOERROR, SAVRES never terminates an operation when it finds a bad block that results in a nonfatal error (see the section "SAVRES Switches").

E.9.1 General SAVRES Error Messages

Table E–10 describes the errors that can occur during SAVRES operations.

Table E–10: General SAVRES Error Messages

Message	Meaning
??? **Bad block in MFD**	SAVRES aborts the operation when it finds a bad block in the Master File Directory (MFD).
%%% **Bad block in SATT.SYS**	There is a bad block in the input SATT.SYS file. SAVRES does not use an input disk that has a corrupt SATT.SYS file.
%%% **Bad data:** **in [MFD]** **[<[P.PN] FILENAME.TYPE>]** **[P,PN]]** **[[GFD, nnn]**	A cluster contained in the described account or file could not be relocated due to a fatal read error on the output volume.
??? **Bad directory for device**	SAVRES discovered a bad link or bad block in the input or output directory structure.
%%% **Bad INIT.SYS on system disk**	SAVRES found a bad block in INIT.SYS while transferring it from the system disk to a SAVE Set volume.
%%% **Can't mix device types in SAVE Set**	A SAVE Set cannot be composed of both tapes and disks. Only all disk or all tape SAVE Sets are legal.
%%% **Can't read bad block file**	SAVRES could not read the bad block file on a RSTS/E disk or a SAVE Set volume.
??? **Can't read SAVE Set volume SATT**	A nonrecoverable I/O error occurred while SAVRES attempted to read the Storage Allocation table on the SAVE Set volume.
%%% **Can't write boot**	The bootstrap could not be written to the first block of the output disk.
%%% **Device hung or write-locked** **Retry (Yes or No)?**	The output device is off line or is write-locked. Correct the condition and try again. Type YES to continue or type NO to abort.
%%% **Device mnemonic must be specified**	Enter the device name along with the unit number.

(continued on next page)

Table E-10 (Cont.): General SAVRES Error Messages

Message	Meaning
%%% **Disk must be DSKINTed**	When you are running SAVRES on line, the output disk must be RSTS/E file-structured or a previous SAVE Set volume. Use the DSKINT option of INIT.SYS to reinitialize the disk. See the *RSTS/E System Installation and Update Guide* for a description of the DSKINT option.
%%% **Duplicate switches**	You specified the same switch more than once.
??? **Error in BADB.SYS allocation**	SAVRES was unable to rebuild the bad block file (BADB.SYS) after completing the current operation.
??? **Fatal output bad block**	A bad block was found at a critical position, preventing SAVRES from completing the current operation.
%%% **Illegal command**	Your response to a prompt was in an unacceptable format.
%%% **Illegal command: no output device specified**	The full function line command you specified did not include an output device specification.
%%% **Illegal density**	The density you specified was in an unacceptable format, was included where none was acceptable, or specified an illegal density value.
%%% **Illegal option**	When booting from a SAVE Set volume, you can use only the RESTORE, IMAGE, and IDENTIFY options.
%%% **Illegal switch**	You used a switch in an unacceptable way.
%%% **Illegal switch combination**	You specified conflicting switches.
%%% **Input disk has only nn% free clusters. Mount it anyway <No>?**	SAVRES discovers the input disk has very few clusters that are not allocated and may find problems copying or restoring the disk. You may want to delete some unnecessary file to increase the chances that SAVRES can complete a successful IMAGE.
%%% **Input disk should be rebuilt**	The disk you mounted is "dirty." The SATT.SYS file may be corrupt. SAVRES requires a valid SATT.SYS; therefore, rebuild the disk using the REFRESH option of INIT or the ONLCLN program. See the *RSTS/E System Installation and Update Guide* for a description of REFRESH. See the RSTS/E Utilities Reference Manual for information about ONLCLN.

(continued on next page)

Table E–10 (Cont.): General SAVRES Error Messages

Message	Meaning
%%% Magtape select error Retry (Yes or No)?	When access to a magnetic tape drive was attempted the selected unit was off line.
%%% No default-type <cr> for more information	You typed a LINE FEED or RETURN in response to a question which has no default.
??? No INIT.SYS on system disk	There is no INIT.SYS in account [0,1] on the system disk.
%%% No previous question - type <cr> for more information	You attempted to return to the previous question but it no longer exists.
%%% Non-existent or hung device	You specified a disk or magnetic tape that does not exist on your system or is not on line. Check your configuration or the device type again and retry.
??? Non-recoverable I/O error	SAVRES found a fatal error condition while trying to recover from a DEVICE HUNG OR WRITE LOCK error. SAVRES aborts the current operation and returns you to the SAV/RES FUNCTION prompt.
%%% Not a valid device	You specified an unacceptable device type in response to the question. This may occur if you specify: • A device other than magnetic tape or disk • A tape rather than a disk (or vice versa) • A wrong disk type on an IMAGE • A device without including a unit number, for example, DK: instead of DK0:
%%% Only nn% of the output disk clusters are available for relocation Mount it anyway <No>?	SAVRES finds there are few free blocks available for relocation on the output disk.
??? Operation aborted at user request	You specified the /NOERROR switch and one of the errors described in Table E-12 occurred, or you typed NO to a RETRY (Yes or No)? question.
%%% Same device specified	You designated the same device for both input and output. Specify the correct output device.
%%% Tape won't respond to density XXXX	SAVRES attempted to set the density of the specified magnetic tape to XXXX but the hardware rejected the request. Specify a density setting that is legal for that device, or use a different device.

(continued on next page)

Table E–10 (Cont.): General SAVRES Error Messages

Message	Meaning
%%% Tape won't respond to any density	The input volume cannot be read at any legal density setting.
%%% This is not the correct volume	The SAVE Set or Pack ID you specified does not match that on the mounted device. A DISMOUNT DEVICE message follows the error message and identifies the volume if it is a SAVE Set or a RSTS/E file-structured disk. This message also occurs if SAVRES requests the next volume of a SAVE Set and you specify a device containing a volume that is not part of the current SAVE Set or is not the correct sequential volume of the current SAVE Set. Mount the correct volume.
%%% This volume has no label	SAVRES did not find a valid SAVE Set label or Pack ID on the specified input volume.
%%% Too many bad blocks on input disk	The number of bad clusters on the input disk exceeds the RSTS/E limit.
%%% Too many bad blocks on output disk	The number of allocated clusters on the original source disk exceeds the number of available clusters on the output disk or the number of bad clusters on the output disk exceeds the RSTS/E limit (161 bad clusters). Use a different disk pack.
%%% Unrecognized switch	You have specified an invalid switch.
??? Volume is full	There was not enough room for the relocation of data clusters due to the blocks found on the output volume.
%%% Wrong size disk	The output volume of an IMAGE or RESTORE must be the same size as the original source. See the discussion of LIKE disks in the section "Definitions of SAVRES Terms."
%%% You will be writing to the booted device	If you write to this device, you need to reboot when you are finished with SAVRES.

E.9.2 Transfer Errors-Fatal and Nonfatal

Transfer errors you may find in a SAVRES operation can be fatal or nonfatal. Fatal transfer errors cause an immediate termination of the current operation. Nonfatal errors, if the /NOERROR switch is not in effect, let the operation continue despite the problem. On the other hand, if you specify the /NOERROR switch and one of the errors in Table E–11 occurs, SAVRES terminates the operation immediately (which indicates the output is not a valid medium). In other words, the nonfatal errors listed and described in Table E–11 become fatal if the /NOERROR is attached to the function response.

Whether or not you specify /NOERROR, the error message ???Volume is full occurs if SAVRES cannot relocate a file due to a bad block on the output device. The relocation does not take place because SAVRES cannot find any additional clusters available or does not find as many free clusters as it needed. This results from too many bad blocks on the output volume.

Table E–11 contains a description of all nonfatal transfer errors that can occur if you do not specify /NOERROR.

Table E–11: SAVRES Nonfatal Transfer Errors

Message and Meaning

%%% Bad block on input, PCN=nnnnnn

A bad block was found on an allocated cluster on the input volume. A copied version of the data in that block might be corrupt.

%%% Bad block on input, output PCN's affected:

A bad block was found on the input SAVE Set volume, which will affect certain pack clusters on the output disk. A copied version of the data in that block might be corrupt.

%%% Bad block on input:
in MFD [PCN=nnnnnn]
in UFD [P,PN] [, PCN=nnnnnn]
in GFD, nnn [, PCN=nnnnnn]
in file [P,PN]FILENAME.TYPE,[PCN=nnnnnn]

A bad block was found on an allocated cluster on the input volume. A copied version of the data in that block might be corrupt.

%%% Bad Compare, PCN=nnnnnn

The data on the input volume did not compare with the corresponding data on the output volume. (See the explanation following this table.)

%%%Bad compare on input:
in MFD[PCN=nnnnnn]
in UFD [P,PN] [, PCN=nnnnnn]
in GFD, nnn [, PCN=nnnnnn]
in file [P,PN]FILENAME.TYPE,[PCN=nnnnnn]

The data on the input volume did not compare with the corresponding data on the output volume. (See the explanation following this table.)

%%% File XXXXXX has been changed to noncontiguous

SAVRES found a bad block in the output area for file XXXXXX. To complete the transfer, one or more blocks of the file were relocated. Consider the consequences of this change and proceed accordingly.

%%% Placed file XXXXXX has been moved from aaaaaa to bbbbbb

A bad block was found in the output area for file xxxxxx. To complete the transfer, one or more blocks of the file were relocated. Consider the consequences of this change and proceed accordingly.

During a verification pass, SAVRES may detect differences between the input and output. If there are differences, SAVRES informs you with two types of messages:

```
%%%Bad compare, PCN=nnnnn

%%%Bad compare in [MFD  [, PCN=nnnnn]]
                  [UFD [p,pn] [, PCN=nnnnn]]
                  [GFD, nnn [, PCN=nnnnnn]]
                  [file [p,pn]filenam.typ [, PCN=nnnnnn]]
```

SAVRES first prints a general error message and then, after some additional processing, prints the second error message telling you more specifically the accounts and/or files in which the bad comparisons occurred.

If you request transfer verification, SAVRES checks (after the initial transfer phase) whether the data it has written matches the corresponding data it read. If there is a difference, SAVRES prints:

```
%%%Bad compare in [MFD  [, PCN=nnnnn]]
```

The SAVRES program transfers only clusters that are marked as allocated in the Storage Allocation Table. During the actual transfer and verify pass, it can tell only which Pack Cluster Number (PCN) it has completed. If /NOERROR is in effect and you are performing an IMAGE copy, SAVRES then tries to pinpoint the problem by scanning the input directory structure before aborting. If you are doing a RESTORE, SAVRES simply aborts. This difference exists because SAVRES must have a complete directory to scan in order to find accounts and/or files that caused the problems. In the case of a RESTORE, the output volume is not yet complete and SAVRES cannot scan it.

If /NOERROR is not in effect, SAVRES stores each PCN that caused a problem in an internal table as it continues with the verify pass. If SAVRES finds more entries than can be stored in the table, it prints the message:

```
"Affected file will not be reported"
```

This means there is no more room to store the information. At the end of the verification pass, SAVRES prints the total number of variations found in the form:

```
%%% nn differences found
```

After completing the transfer, SAVRES scans the input (IMAGE) or (RESTORE) volume directories to find the accounts and/or files in which the bad comparisons occurred. Then, it displays one or more of the following messages:

```
%%%Bad compare in [MFD  [, PCN=nnnnn]]
                  [UFD [p,pn] [, PCN=nnnnn]]
                  [GFD, nnn [, PCN=nnnnnn]]
                  [file [p,pn]filenam.typ [, PCN=nnnnnn]]
```

Appendix F

Device Testing Package (DEVTST)

The Device Testing Package (DEVTST) exercises hardware on the RSTS/E system during normal time-sharing operations. You can use DEVTST to simulate hardware errors on the system.

F.1 Introduction to DEVTST

DEVTST is a package of programs that provides the customer, System Manager, or DIGITAL Field Service personnel with tests that verify the reliability of RSTS/E-supported hardware. Execution of any DEVTST program is limited to users with WREAD privilege. As distributed, DEVTST is in TEST$: (by default, [0,5]).

This appendix explains operational data, conventions, complete device testing information, and error messages. After execution of individual DEVTST exercisers, you should check the System Error Log for additional errors; for more information on the System Error Log, see Chapter 10.

The DEVTST exercisers are tests that can be initiated whenever you suspect hardware trouble. These device exercisers generate excess activity in an attempt to make the hardware fail during operation.

Use the DEVTST package in conjunction with the RSTS/E error package (see Chapter 10). The Error Logging facility provides the means for determining detailed information on hardware errors. In addition, the DEVTST programs return error messages that may be useful in determining the failure; however, these messages are not as specific as the information provided by the System Error Log.

The following conventions are used throughout the DEVTST package dialogue:

< > Indicates the default answer.

( ) Shows all possible replies.

[ ] Gives an optional reply; for example, ASC [II] means enter ASC, ASCI, or ASCII.

<ALT MODE> Either key moves the program back one question. If you

or <ESC> press either key as a response to the first question, DEVTST will stop execution and return to the RSTS/E command level.

<Ctrl/Z> Pressing Ctrl/Z causes the program to stop execution and return to RSTS/E command level. Ctrl/Z is detected only when the system checks for new input.

<Ctrl/C> Pressing Ctrl/C causes the program to stop execution immediately and return to a command level.

<RET> Use the Return key to terminate the input line.

Detach <NO> All of the device exercisers have the ability to detach; this allows you to run an exerciser while keeping your terminal free for other operations.

-1 Negative one, when used as a value for number of iterations, pages, or lines, tells the program to run indefinitely, or until you press Ctrl/C.

F.2 DEVTST Operating Procedures

To run a DEVTST program, type:

```
$ RUN TEST$:program name
```

For example:

```
$ RUN TEST$:KBEXER
```

Table F–1 lists the program names and the hardware they test.

Table F–1: DEVTST Programs

Program	Hardware Tested
CPEXER	CPU
CPUTST	CPU
DSKEXR	File-structured disk
DSKSEK	Non-file-structured disk
DXEXER	Diskette
KBEXER	Terminal
LPEXER	Line printer
MTEXER	Magnetic tape

Table F–2 gives a complete list of RSTS/E supported hardware, including designators and the associated devices.

Table F–2: RSTS/E Device Designators

Device Designator	Device
DK:, DL:, DM:, DP:, DR:, DB:, DU:, SY:	RSTS/E public disk structure
SY0:	System disk (the unit that was bootstrapped)
DV0:	Virtual disk, unit 0 only
DK0: to DK7:	RK05/RK05F disk cartridge units 0 to 7
DL0: to DL3:	RL01/RL02 disk cartridge units 0 to 3
DM0: to DM7:	RK06/RK07 disk cartridge units 0 to 7
DR0: to DR7:	RM02/RM03/RM05/RM80 disk units 0 to 7
DB0: to DB7:	RP04/RP05/RP06 disk pack units 0 to 7
DU0: to DU15:	RA60, RA80, RA81, RC25, RD51, RD52, RD53, RX50 disk units 0 to 15
CR:	CR11 punched or CM11 mark sense card reader
CD:	CD11 punched card reader
MT0: to MT7:	TE10/TU10/TS03 magnetic tape units 0 to 7
MS0: to MS3:	TS11/TSV05/TK25/TU80 magnetic tape units 0 to 3
MM0: to MM7:	TE16/TU16/TU45/TU77 magnetic tape units 0 to 7
MU0: to MU7:	TK50 magnetic tape units 0 to 7
LP0: to LP7:	Line printer units 0 to 7

F.3 DEVTST Dialogue Errors

Table F–3 lists the error messages that can occur in DEVTST programs.

Table F–3: DEVTST Error Messages

Message and Meaning

%Warning: <device name> is a logical device: <device name> will be used.

DEVTST determines that the device name given is a logical namefor another device. DEVTST performs a logical devicetranslation and the translated device name is used.

?Illegal device: <device name> specified.

The device name given by the user is either incorrect or is alogical device name that cannot be translated (for example,incorrect characters or the wrong type of device). Retry theoperation with the correct device name.

?OPEN failure on <device name>: (Followed by an error message from Appendix A of the ***RSTS/E System Installation and Update Guide.)***

This error message can mean one of the following:

- The device specified does not exist
- No volume was mounted on that device
- Another user has assigned the device
- The volume on the device has not been formatted
- The device is not on-line or the device is not write-enabled

To resolve: retry the operation using the correct option as determined by the error message received.

?Illegal number of iterations selected.

Retry the operation using the correct number of iterations.

F.4 CPEXER

The central processor exerciser is designed to put a heavy load on the Central Processing Unit (CPU). CPEXER runs compute bound for bursts and then sleeps for five seconds so the average processor loading does not appreciably degrade the performance of the system. CPEXER was specifically designed to test the Floating Point Processor (FPP) option, but also serves to verify general CPU integrity and the Extended Instruction Set (EIS) operation. There is also a test of the PEEK function for kernel addresses 0 through 22000.

The next section itemizes the tests performed by CPEXER. In most of the tests, CPEXER compares the results to known correct values. Two heavily compute-bound tests are included to verify consistent results of duplicate calculations. Finally, CPEXER includes tests that verify FPP divide by zero trap, integer divide by zero trap, integer conversion error, and EIS operation.

F.4.1 CPEXER Tests

The CPEXER program includes the following 13 CPEXER tests:

Test	Description
SIN(X)	Uses SIN(X) extended function
SIN(X)	Uses polynomial approximation to SINE function
LOG(X)	Uses LOG(X) extended function
EXP(X)	Uses EXP(X) extended function
SQR(X)	Uses SQR(X) extended function
SQR(X)	Uses Newton-Raphson method to determine the square root
LOG(EXP(X))	Compute-bound test
ATN(TAN(X))	Compute-bound test
A=1.0/0.0	Verifies FPP divide by zero trap
A%=1%/0%	Verifies integer divide by zero trap
A%=60000	Verifies integer conversion error
INT(40.6621*100+0.5)/100=40.66	Verifies EIS operation
PEEK(X)	Verifies operation of PEEK function

F.4.2 CPEXER Dialogue

The following is an example of the CPEXER dialogue:

```
$ RUN TEST$:CPEXER

CPEXER V9.0 RSTS V9.0 EDERE

How many minutes to run <15>? 1

Detach <No>? NO

11.9 seconds of CPU time used in 1 pass.

CPEXER finished at 03:36 PM

$
```

NOTE

The CPEXER program will build correctly only if the version of BASIC-PLUS or BASIC-PLUS-2 includes TRIG and LOG. These mathematical functions must be included during system installation of BASIC-PLUS or the installation of BASIC-PLUS-2.

F.4.3 CPEXER Errors

This program reports four types of errors:

1. Instructions failed to execute.
2. Errors failed to trap.
3. Computation errors.
4. Computation consistency checks failed.

Check the System Error Log for reported errors. These errors may indicate hardware problems.

When type one errors (instruction errors) occur, CPEXER prints a two-line message: the first line gives the test that was being performed and the exact error message, and the second line indicates that the test is continuing. For example:

```
?CPEXER PEEK failed at memory location m.
-(Error Message)

Continuing......
```

where:

(Error Message) is the RSTS/E error message from Appendix A of the *RSTS/E System Installation and Update Guide.*

m is the memory location at which the PEEK function failed.

Examples of type two errors (trap failure) are:

```
?CPEXER Floating point divide by 0 didn't trap

Continuing.....

?CPEXER Integer divide by 0 didn't trap

Continuing.....

?CPEXER Forced integer conversion didn't trap

Continuing.....

?CPEXER (INT(40,6621*100=0.5))/100< >40.66

Continuing.....
```

Type three errors (computation errors) give the type of test (SIN,PLYNOM, LOG, EXP, SQR, or NEWTON SQR), the error in the computed value, and the location of the possible error. For example:

```
?CPEXER - SIN test failed for the following cases:
?CPEXER - Computed value of SIN(40) was .643, should be .64
?CPEXER -  CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing ...

?CPEXER - PLYNOM SIN test failed for the following cases:
?CPEXER - Computed value of PLYNOM SIN (40) was .643, should be .64
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...

?CPEXER - LOG test failed for the following cases:
?CPEXER - Computed value of LOG (140) was 4.942, should be 4.939
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...

?CPEXER - EXP test failed for the following cases:
?CPEXER - Computed value of EXP (4) was 54.598, should be 54.599
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing ...
```

```
?CPEXER - SQR test failed for the following cases:
?CPEXER - Computed value of SQR (500) was 22.361, should be 22.362
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...
```

```
?CPEXER - NEWTON SQR test failed for the following cases:
?CPEXER - Computed value of NEWTON SQR (500) was 22.361, should be 22.
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...
```

Type four errors (consistency failure) in the CPEXER are shown in the following examples:

```
?CPEXER - EXP/LOG Consistency check failed for the following cases:
?CPEXER - EXP(LOG(...( 1 )))))))))) <> EXP(LOG(...( 1))))))))))
?CPEXER - EXP(LOG(...( 4 )))))))))) <> EXP(LOG(...( 4))))))))))
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...
```

```
?CPEXER - ATN/TAN Consistency check failed for the following cases:
?CPEXER - ATN(TAN(...(-1.5 )))))))))) <> ATN(TAN(...(-1.5 ))))))))))
?CPEXER - CPU,EIS,FIS,FPP,MUL,DIV or SHIFT Error
Continuing...
```

F.5 CPUTST

The CPUTST dialogue first asks how many minutes you want the test to run. The program then:

1. Checks for expiration of this request time at eight strategic points during each run sequence
2. Clears and/or sets three 2-dimensional arrays to specified values
3. Transposes Matrix A and places the result in Matrix B
4. If no errors are found, returns to Step 2
5. If an error is found, reports the error and aborts the test

The program repeats this process until the run time expires, at which time it prints the CPU time in seconds of expended job time.

F.5.1 CPUTST Dialogue

The following is an example of CPUTST dialogue:

```
$ RUN TEST$:CPUTST
CPUTST V9.0 RSTS V9.0 EDERE

How many minutes to run <15>? 1

Detach <No>? NO

6.9 seconds of CPU time used.

CPUTST Finished at 10:59 AM

$
```

F.5.2 CPUTST Errors

This device exerciser reports errors as shown in the following examples:

```
?CPUTST-Error in matrix transposition routine.
```

```
?(Error message) at line mm in CPUTST V9.0.
```

where:

(Error Message)	is the RSTS/E message in Appendix A of the *RSTS/E System Installation and Update Guide* for the error that occurred.
mm	is the BASIC-PLUS line number in CPUTST where the error occurred.

Check the System Error Log for reported errors. These errors may indicate hardware problems.

If your hardware does not include a floating point processor (FPP), submit a Software Performance Report (SPR) to DIGITAL whenever errors appear.

F.6 DSKEXR

This disk exerciser tests the normal operation of all file-structured disks. It is possible to run several copies of DSKEXR in order to test several drives simultaneously or to put a heavier load on any single drive. The disk must be physically and logically mounted. DSKEXR begins by asking questions to determine the disk drive to be tested and the number of test iterations to be performed.

After this dialogue, the exerciser:

1. Opens and extends a temporary file in the current user's account on the specified disk to a predetermined size that is dependent upon disk size and the number of free blocks
2. Loads a pattern buffer with one of four patterns (all 1's and 0's or a pattern of alternating 1's and 0's)
3. Writes the file
4. Reads each block and compares it for each of the patterns.

Upon completion of all iterations for each drive you specify, DSKEXR prints a status report.

NOTE

> This write operation does not use current disk files to read and write. The temporary file will always be deleted after use, regardless of how the DSKEXR program is terminated. Be sure to turn off data caching before you run this test.

F.6.1 DSKEXR Dialogue

The following is an example of DSKEXR dialogue:

```
$ RUN TEST$:DSKEXR
DSKEXR V9.0 RSTS V9.0 EDERE
Disk drive to test? DB1:

Number of test iterations <8>? 1

Detach <No>? NO

No errors detected on _DB1:

$
```

F.6.2 DSKEXR Errors

DSKEXR reports errors as shown in the following examples:

```
?(Error message) error at line mm in DSKEXR V9.0

?nn errors detected on DBn.
```

where:

(Error message) is the RSTS/E message in Appendix A of the *RSTS/E System Installation and Update Guide* for the error that occurred.

mm is the line number where the error occurred in the DSKEXR program.

nn is the number of errors detected by DSKEXR.

When DSKEXR detects errors, do the following:

1. Check to make sure that the volume is physically mounted and write-enabled
2. Check to make sure that the volume is logically mounted. This may be verified by the use of the SHOW DISKS command. If the disk is not mounted use the MOUNT command (see Chapter 15).

CAUTION

Note the following to avoid damage to the system:

- Packs mounted as unlocked public disks will be available to other system users for creation and storage of their files. Subsequent removal of a public disk may result in the elimination of these user files from the system. For this reason, DIGITAL recommends the use of a private pack if a nonsystem disk is to be tested.
- Never mount a potentially bad disk on another drive until you are sure that the drive and pack are free of physical damage.

Persistent errors may be due to bad blocks on the pack. Any potentially bad blocks will be flagged in the System Error Log following unsuccessful access, to prevent further access to the INIT option.

F.7 DSKSEK

This disk exerciser tests the normal operation of non-file-structured disks by randomly reading blocks from the disk. It is possible to run several copies of DSKSEK in order to test several drives simultaneously or to put a heavier load on any single drive. If the disk being tested is a mounted RSTS/E file-structured disk, DSKSEK will not read the known bad blocks. If an error occurs on a read operation, DSKSEK reports it will be reported by DSKSEK and logs it into the System Error Log.

NOTE

> This is a read-only operation. Turn off data caching before you run this test.

F.7.1 DSKSEK Dialogue

The following is an example of DSKSEK dialogue:

```
$ RUN TEST$:DSKSEK
DSKSEK V9.0 RSTS V9.0 EDERE

Disk drive to test? DB0:

Number of test iterations <30>? 1

Detach <No>? NO

No bad block detections in 1000 sets from _DB0:

$
```

F.7.2 DSKSEK Errors

Errors detected by DSKSEK are displayed as follows:

```
?(Error message) error at line mm in DSKSEK V9.0.
```

where:

(Error Message) is the RSTS/E message in Appendix A of the *RSTS/E System Installation and Update Guide*.

mm is the BASIC-PLUS line number where the error occurred in the DSKSEK program.

When DSKSEK detects errors, do the following:

1. Check to see if the volume on the testing device is physically mounted.
2. If it is, mount another scratch volume on the drive and retry the operation.

CAUTION

> Never mount a potentially bad disk on another drive until you are sure that the drive and pack are free of physical damage.

3. If the test continues to fail, check the System Error Log to see if bad blocks are recorded.

4. If bad blocks are recorded, add them to the bad block list on the disk using the REFRESH BADS suboption of INIT. This can only be done if the disk has been initialized (DSKINTed) as a RSTS/E disk. (See the *RSTS/E System Installation and Update Guide* for information on BADS.)
5. If the problems continue, get a complete listing of the System Error Log (using the ERRDIS program).

F.8 DXEXER

This exerciser tests the normal operation of the diskette controller and the specified diskette drives. DXEXER starts by opening the floppy and writing 420 blocks of floating point numbers to it. These are written on the drive being tested, then read back and verified. The program then displays the number of words written and read, and the number of errors detected.

CAUTION

> DXEXER destroys any information previously contained on the diskette used for this test.

F.8.1 DXEXER Dialogue

The following is an example of DXEXER dialogue:

```
$ RUN TEST$:DXEXER
DXEXER V9.0 RSTS V9.0 EDERE

Warning: This exerciser will destroy data on the tested flexible
diskette.

Flexible drive to test? DX0:

Number of iterations <20>? 2

Detach <No>? NO

Flexible diskette DXEXER I/O data

Device  Reads    Words  Writes  Words
_DX0:             300            76.8 K     300          76.8 K

DXEXER Finished at 10:53 AM

$
```

F.8.2 DXEXER Errors

DXEXER shows the number of errors detected during operation as shown in this example:

```
?mm errors on DXn during iteration nn.
```

where:

mm is the number of errors detected by the DXEXER program.

DXn is the name of the device being tested.

nn is the number of the iteration during which the error occurred.

When DXEXER detects errors, do the following:

1. Check to see if the flexible diskette on the testing device is physically mounted and the door closed.
2. If it is, mount another scratch flexible diskette on the drive and retry the operation.

F.9 KBEXER

The keyboard exerciser (KBEXER) is used to test local and remote terminals connected to the RSTS/E system. The three terminal tests are as follows:

- The Space Test—Verifies that the terminal carriage will return reliably from any position
- The ASCII Pattern Test—Verifies that the terminal will print the standard ASCII character set in all print positions
- The Repeat Test—Repeats everything typed, one line at a time

NOTE

You can terminate the tests at any time by pressing Ctrl/C. You can obtain HELP by typing HELP in answer to any dialogue question.

F.9.1 KBEXER Dialogue

The following is an example of KBEXER dialogue:

```
$ RUN TEST$:KBEXER
KBEXER V9.0 RSTS V9.0 EDERE

Keyboard to test <_KB68:>? <RET>

Test (SPA[CE], ASC[II], WOR[ST], REP[EAT], HEL[P])? ASC

Number of test iterations <8>? 1

Detach <No>? NO

  *** ROTATING ASCII CHARACTERS TEST ***

#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !"#$%&
"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !"#$%
!"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !"#$
 !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !"#
_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !"
^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_ !
]^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
\]^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^_
[\]^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]^
Z[\]^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\]
YZ[\]^_ !"#$%&'()*+,-./0123456789:;<=>? @ABCDEFGHIJKLMNOPQRSTUVWXYZ[\
 .
 .
 .
KBEXER Finished at 10:33 AM

$
```

F.9.2 KBEXER Errors

Errors detected by KBEXER appear as errors in the keyboard test patterns. Visually check the patterns for errors.

F.10 LPEXER

This exerciser prints a rotating pattern of ASCII characters on a specified number of pages and is used to test any line printer.

F.10.1 LPEXER Dialogue

The following is an example of LPEXER dialogue:

```
$ RUN TEST$:LPEXER
LPEXER V9.0 RSTS V9.0 EDERE

Line printer to test? LP0:

Number of pages to output <4>? 1

Detach <No>? NO

LPEXER finished at 11:38 AM

$
```

F.10.2 LPEXER Errors

Problems found on the line printer appear as errors in the print-out of ASCII characters. Visually check the line printer listing and if an error message is printed by LPEXER, check to make sure that the line printer selected is on-line and not out of paper.

F.11 MTEXER

The magnetic tape exerciser checks normal operation of the TM11 or TS11 magnetic tape controller, the RH11/RH70/TM02/TM03 controller, and the specified number of 9-track TE10, TU10, TS03, TS11, TU16, TE16, TU45, TU77, TU80, TU81, TK25, or TK50 drives. MTEXER allows you to select the drive to be tested, the length of tape to be written and the number of iterations to be performed. For each iteration, MTEXER initializes the tape, sets the drive to its maximum legal density, opens a file and writes data until the specified length of tape has been used. MTEXER then rewinds the tape, opens the file for input, reads and verifies the data. If errors are detected, MTEXER prints a count of the number of bytes found to be incorrect and then proceeds to the next iteration.

The data pattern used is a worst case NRZI pattern for 9-track drives. The pattern is loaded into a 512-byte buffer, and successively greater numbers of records are written to the magtape until the requested length of tape has been written. Furthermore, the pattern base varies with the iteration number so the contents of the pattern buffer also varies on successive iterations.

CAUTION

This exerciser overwrites data contained on the magnetic tape used for the test.

F.11.1 MTEXER Dialogue

The following is an example of MTEXER dialogue:

```
$ RUN TEST$:MTEXER
MTEXER V9.0 RSTS V9.0 EDERE

Warning: This exerciser will destroy data on the tested magtape.

Magtape drive to test? MT2:

Drive type is TE16/TU45/TU77

Density (800 or 1600) <800>? <RET>

Number of feet to test <100>? <RET>

Number of iterations <2>? 1

Detach <No>? NO

Magtape MTEXER I/O Data

Device  Reads  Words  Writes  Words
_MT2:            387            99.072 K   387          99.072 K

No errors detected on _MT2:

$
```

The details of the dialogue vary with the type of drive used. For instance, if you use a drive with only one possible density, the dialogue does not include a density question like the one in the example. Also, the question about the number of feet does not reflect the actual number of feet on a TK50 tape.

F.11.2 MTEXER Errors

Magtape errors are reported in these formats:

```
?Magtape select error at line nn in MTEXER V9.0.

?MTEXER-Selection error occurred.
```

where:

nn   is the line number in MTEXER at which the error occurred.

When MTEXER detects errors, do the following:

1. Check to see if the volume on the testing device is physically mounted, write-enabled, and assigned to you.
2. If it is, mount another scratch volume on the drive and retry the operation.
3. Check to make sure that the tape path and heads are free of oxide.

Index

A

- ABORT command
  - OPSER, A–45
  - SPOOL, A–26, A–35
- /ABORT qualifier
  - STOP/QUEUE/MANAGER command, 9–45
- Accessing entries
  - entry number, 9–7
  - entry specification, 9–7
- Account
  - associated with system logical name, 12–8
  - nonuser, 7–1
  - programming error in privileged, 2–5
  - user, 7–1
- Account attribute
  - captive flag, 7–4
  - dial-up flag, 7–4
  - network flag, 7–4
- /ACCOUNTING qualifier
  - RESTORE command, 16–2
- /ACCOUNTING_DATA qualifier
  - SHOW ACCOUNT command, 7–19
- Account management
  - GACNT privilege, 6–3
  - WACNT privilege, 6–7
- Account management, DCL commands, 7–1t
- /ACCOUNT_DATA qualifier
  - BACKUP command, 16–5
  - RESTORE command, 16–13
- Action requests
  - BATCH, A–41
  - OPSER, A–10
  - table of, A–10
- /ADDRESS qualifier
  - CREATE/VIRTUAL_DISK command, 11–15
  - INSTALL/LIBRARY command, 12–23
  - INSTALL/RUNTIME_SYSTEM command, 12–26
- /ADVANCED_VIDEO qualifier
  - SET TERMINAL command, 8–16
- AFTER state, processing entries, 9–6
- /ALIGN switch, SPOOL, A–37
- ALLOCATE command, 11–13
- /ALLOCATED qualifier
  - SHOW DEVICE command, 11–28
- /ALL qualifier
  - BROADCAST command, 13–9
  - DEALLOCATE command, 11–16
  - DEASSIGN/QUEUE command, 9–22
  - DELETE/COMMAND/SYSTEM command, 12–20
  - FORCE command, 13–12
  - LOAD/OVERLAY command, 14–23
- /ALL qualifier (cont'd.)
  - SET CACHE command, 11–20
  - SET QUEUE command, 9–32
  - SET SERVER command, 9–35
  - SHOW ACCOUNT command, 7–19
  - SHOW DEVICE command, 11–28
  - SHOW FILE/SYSTEM command, 12–38
  - SHOW LOGICAL/SYSTEM command, 12–41
  - SHOW LOGICALS command, 12–40
  - SHOW PRINTER command, 11–30
  - SHOW SERVER command, 9–37
  - SHOW TERMINAL command, 8–30
  - UNLOAD/OVERLAY command, 14–32
- /ALT_MODE qualifier
  - SET TERMINAL command, 8–16
- ANALYS
  - CRASH.SYS, 12–16
  - crash dump data, 10–23
  - crash error code, 10–23t
  - create error logging file, 10–1
  - dialogue questions, 10–22t
  - output, 10–23
  - run, 10–21
  - running after crash, 3–9
  - START.COM commands, 10–23
  - use, 10–21
- /ANSI qualifier
  - SET TERMINAL command, 8–16
- /ANSWERBACK qualifier, 14–21
  - SET SYSTEM command, 14–25
  - SET TERMINAL command, 8–16
- ANSWER command
  - BATCH, A–41
  - OPSER, A–11
- ASSIGN/QUEUE command, 9–20
- ASSIGN/SYSTEM command, 12–9, 12–17
- ASSIGN option, SPOOL, A–24
- ATTACH command, 13–8
- Attributes, printer forms, 9–13
- /AUTOBAUD qualifier
  - SET TERMINAL command, 8–1, 8–16
  - TERSPD.SYS file, 8–2
- Automatic restart
  - after power fail, 2–6
  - procedures, 2–2
  - RSTS/E system, 2–2
  - system initialized, 2–5
- Auxiliary run-time system, 12–3
  - adding, 12–4
  - creation of, 12–4
  - removing, 12–4
  - START.COM, 12–4

B

/BACKSPACE qualifier
  SET PRINTER command, 11–23
  START/SERVER command, 9–42
BACKUP, comparison with SAVRES, E–1
BACKUP command, 16–3 to 16–11
  quick reference table, 16–10t
Backup sets
  restrictions, 16–4
BADB.SYS file
  adding bad blocks, 10–8
Bad blocks
  adding, caution, 10–8
  adding to BADB.SYS, 10–8
  criteria, 10–7
  during RESTORE, E–8
  ERRDIS example, 10–8
  ERRDIS prints list, 10–8
  ERRDIS report, 10–7
  list of potential, 10–4
  report content, 10–7
  SAVRES file changes, E–24
  SAVRES output, E–24
Base priority
  SET JOB/PRIORITY command, 6–6
  TUNE privilege, 6–6
  UU.PRI system directive, 6–6
BASIC-PLUS
  keyboard monitor, 1–2
  run-time system, 12–3
BATCH, A–1
  ANSWER command, A–41
  answering action requests, A–41
  communication paths, A–5
  CONTINUE command, A–42
  default conditions for, A–38
  default receiver identification, A–38
  device type designators, A–41t
  dismount a volume, A–42
  error checking, A–38
  interrupt commands, A–40t
  logical device name, A–38
  modules, A–37
  mount requests, A–41
  OPSER communicate with, A–40
  OPSER package, A–37
  overview, A–5
  processing command file, A–41
  processor, A–37
  pseudo keyboard, A–38
  requests for operator action, A–41
  run, A–37
  startup options, A–39t
  startup procedures, A–41
BATCH.DCD file
  command decoding, A–5
Batch jobs, exit status messages, 9–12
/BATCH qualifier
  INITIALIZE/QUEUE command, 9–26
  SET SERVER command, 9–35
  SHOW SERVER command, 9–37
Batch servers
  CPU limits, 9–12
  Print/Batch Services (PBS), 9–2, 9–11
  pseudo keyboards, 9–12
  time limits, 9–12
Batch servers (cont'd.)
  wildcard names, 9–11
/BELL qualifier
  BROADCAST command, 13–9
Bit 0, setting, 2–5
/BLOCK_SIZE qualifier
  BACKUP command, 16–5
Bootable medium, create with SAVRES, E–18
Bootstrap
  RSTS/E after system halt, 2–2
  RSTS/E into memory, 2–1
Bootstrap procedure, 2–1
/BREAK qualifier
  SET TERMINAL command, 8–17
/BRIEF qualifier
  BACKUP command, 16–5
  RESTORE command, 16–13
  SHOW ACCOUNT command, 7–20
  SHOW SERVER command, 9–37
  SHOW TERMINAL command, 8–30
BROADCAST command, 13–3, 13–9
  SEND privilege, 6–5
/BROADCAST qualifier
  SET TERMINAL command, 8–17
/BROADCAST qualifier, permits receiving broadcasts, 8–6
Bucket size
  cache, 11–10
  RMS, 11–11
Buffer status report, SHOW BUFFERS command, 12–15
/BUFFER_QUOTA qualifier
  SET TERMINAL command, 8–17
/BUFFER_SIZE qualifier
  BACKUP command, 16–5
  RESTORE command, 16–13
Burst pages, A–25
BYE command, CCL, 3–8

C

Cache
  clusters in, 11–8
  data block, 11–7
  size of, 11–8
  space for, 11–8
  updated by the monitor, 11–8
Cache cluster
  eligible for replacement, 11–9
  last block in, 11–9
  size of, 11–8, 11–19
  specify size of, 11–20
/CACHE qualifier
  SET FILE command, 12–35
Caching
  checks by monitor, 11–19f
  directory, 11–7
  efficiency of, 11–10
  guidelines, 11–10, 11–11
  MODE checks, 11–10
  optimize directory, 11–11
  override UFD entry, 11–10
  random mode, 11–8
  RMS indexed file, 11–11
  sequential mode, 11–9
  types, 11–7
  UFD entry marked for, 11–9

Caching Control, DCL commands, 11–1t
Captive account flag, 7–4
/CAPTIVE qualifier
  CREATE/ACCOUNT command, 7–6
  SET ACCOUNT command, 7–14
Catastrophic error, 2–2, 2–5
  automatic recovery, 2–4, 2–5
  cause, 2–4, 2–5
  handling, 2–5
  recovery, 2–5
CCL (Concise Command Language)
  adding CCL commands, 12–11
  adding in sequence, 12–11
  BYE command, 3–8
  command definition, 3–8
  defined at start of time sharing, 12–10
  definition, 1–1
  error checking, 12–11
  HELLO command, 3–8
  listing, 12–12
  PLEASE, A–44
  redefining, 3–8
  removing CCL commands, 12–13
  run RSTS/E programs, 12–10
  use of small buffers, 3–8
CCL command, definition, 12–10
CHANGECONSOLE command
  OPSER, A–13
CHARS.QUE file, A–35
  character generation, A–4
CLOSE/QUEUE command, 9–21
/CLOSE qualifier
  DETACH command, 13–11
  INITIALIZE/QUEUE command, 9–26
Cluster size
  cache and pack, relationship, 11–9
  cache and RMS bucket size, 11–10
  disk type, 15–8t
/CLUSTERSIZE qualifier
  SET CACHE command, 11–20
/CLUSTER_SIZE qualifier
  CREATE/ACCOUNT command, 7–7
  INITIALIZE command, 15–7
/132_COLUMNS qualifier
  SET TERMINAL command, 8–16
Command file
  QUEMAN startup, A–19
  system startup, 3–2
Command level
  system, 1–11
  terminal at, 1–11
Command line, full function (RESTORE), E–18
/CONFIRM qualifier
  REMOVE/JOB command, 13–16
Console terminal
  ability to log on to, 13–2
  change, 13–2
CONTINUE command
  BATCH, A–42
Controlled access
  privilege, 1–12
/CONTROL qualifier
  SET TERMINAL command, 8–17
/CONTROLS qualifier
  INITIALIZE/SERVER command, 9–29
  SET PRINTER command, 11–23
  SET SERVER command, 9–35
CPEXER program
  CPU exerciser, short, F–4
  CPU integrity test, F–4
  EIS operation test, F–4
  example dialogue, F–5
  FPP test, F–4
  LOG feature requirement, F–5
  PEEK function test, F–4
  tests performed, F–5
  TRIG feature requirement, F–5
  type four errors, F–7
  type one errors, F–6
  type three errors, F–6
  type two errors, F–6
CPU (Central Processing Unit)
  determine the state of, 2–3
CPU exerciser, F–4
CPU integrity test
  CPEXER program, F–4
CPUTST program, F–7
  errors, F–8
  example dialogue, F–7
/CPU_LIMIT qualifier
  INITIALIZE/QUEUE command, 9–26
  SET QUEUE command, 9–33
Crash
  analyzing system, 10–21
  causes of system, 2–5
  documenting system, 10–21
  error code, 10–23
  handling of system, 2–5
  memory dump, annotated version, 10–22
  OPSER starts after system, A–14
  random system, 2–5
  running QUEMAN after system, A–20
  system, 2–2, 2–5
CRASH.SYS file, 10–1, 10–21
  crash dump, 12–16
Crash file
  CRASH.SYS, 12–16
  information from, 10–1
/CRC qualifier
  BACKUP command, 16–6
CREATE/ACCOUNT command, 7–5
CREATE/SERVICE/LAT command, 8–9
CREATE/VIRTUAL_DISK command, 11–12, 11–14
Create access
  GWRITE privilege, 6–3
  WWRITE privilege, 6–7
/CREATED qualifier
  BACKUP command, 16–5
  RESTORE command, 16–13
/CRFILL qualifier
  SET TERMINAL command, 8–17
/CR_PREFIX qualifier
  SET PRINTER command, 11–24
Ctrl/C
  do not use in QUEMAN, A–16
  FORCE, 13–4
  SAVRES dialogue, E–5
  typed at terminal, 1–11
Ctrl/Z
  FORCE command, 13–4
  SAVRES dialogue, E–5
CUSP (Commonly Used System Program)
  definition, 1–2

D

Data access, cached, 11–8
Data block
  cache, 11–7
  retrieval pointer, 11–2
Data caching
  cache clusters used, 11–20
  control, 11–7
  disabling, 11–20
  enable on the system, 11–18
  guidelines, 11–9
  memory access, 11–7
  minimum residency time, 11–11
  MODE values, 11–7
  monitor directives, 11–7
  read operations, 11–7
  restriction with DSKSEK program, F–10
  RMS files, 11–11
  support for, 11–7
  write operations, 11–7
  XBUF allocation, 11–8
/DATA_MAXIMUM qualifier
  SET CACHE command, 11–20
/DATE qualifier
  INITIALIZE command, 15–7
DATES privilege
  system clock, 6–3
/DATE_FORMAT qualifier
  SET SYSTEM command, 14–26
DCL (Digital Command Language)
  definition, 1–1
  keyboard monitor, 1–2
DCL (DIGITAL Command Language)
  caution when defining, 3–8
  CCL commands, 3–8
DCL command descriptions
  account management, 7–5 to 7–20
  BACKUP, 16–3 to 16–11
  caching control, 11–12 to 11–30
  device management, 11–12 to 11–30
  disk handling, 15–3 to 16–1
  job management, 13–7 to 14–1
  LAT management, 8–9 to 9–1
  managing logical names, 12–17 to 13–1
  managing system files, 12–17 to 13–1
  Print/Batch Services Package (PBS), 9–19 to 9–47
  RESTORE, 16–11 to 16–18
  system functions, 14–22 to 15–1
  tape handling, 15–3 to 16–1
  terminal management, 8–9 to 8–29, 9–1
DCL commands
  account management, 7–1t
  disk handling, 15–1t
  job management, 13–1t
  logical names, managing, 12–1t
  Print/Batch Services Package (PBS), 9–1t
  system files, managing, 12–1t
  system functions, 14–1t
  tape handling, 15–1t
DEALLOCATE command, 11–16
DEASSIGN/QUEUE command, 9–22
DEASSIGN/SYSTEM command, 12–18
  logical name, 12–10
DECnet/E
  crash analysis, 10–23

DECnet/E (cont'd.)
  disabling
    during shutdown, 4–4, 4–6
    logins, 4–3
  EVTLOG
    program, 4–6
    shutdown phase, 4–8
  interrupt shutdown, 4–6
  Network Services Protocol file, 12–15
  resetting shutdown status, 4–6
  shutdown, 4–6
    example, 4–12
    phase, 4–5
  use of NCP, 4–6
/DEFAULT qualifier
  INITIALIZE/QUEUE command, 9–26
  SET QUEUE command, 9–33
Default queues
  Print/Batch Services, 9–3
DEFINE/COMMAND/SYSTEM command, 12–10, 12–11, 12–19
/DELETABLE qualifier
  SET FILE command, 12–35
DELETE/ACCOUNT command, 7–10
DELETE/COMMAND/SYSTEM command, 12–13, 12–20
DELETE/QUEUE command, 9–23
DELETE/SERVER command, 9–24
DELETE/SERVICE/LAT command, 8–11
DELETE/VIRTUAL_DISK command, 11–12, 11–17
Delimiter
  keyboard, 13–6
/DELIMITER qualifier
  SET TERMINAL command, 8–17
/DENSITY qualifier
  BACKUP command, 16–6
  INITIALIZE command, 15–12
  SET SYSTEM command, 14–26
/DENSITY switch, SAVRES, E–6
Description block, run-time system, 12–4, 12–5
DETACH command, 13–7, 13–11
  OPSER, A–13
  QUEMAN, A–16
Detached job
  check with SYSTAT, 14–8
  SHOW JOBS/DETACHED command, 14–8
Detached-job quota, 5–3
/DETACHED_JOB_QUOTA qualifier
  CREATE/ACCOUNT command, 7–7
  SET ACCOUNT command, 7–14
Device designators, F–3t
  table of BATCH, A–41t
Device management, DCL commands, 11–1t
DEVICE privilege
  restricted device, 6–3
/DEVICE qualifier
  INSTALL/SWAP_FILE command, 12–28
  SHOW COUNTERS/LAT command, 8–24
  SHOW SESSIONS/LAT command, 8–27
Device sizes, disk type, C–2t
Device testing package
  definition, F–1
Device Testing Programs
  see DEVTST programs
/DEVICE_TYPE qualifier
  SET PRINTER command, 11–24
  SET TERMINAL command, 8–18

- DEVTST programs, F–2t
  - conventions, F–2
  - error messages, F–4t
  - how to run, F–2
- /DFLENGTH switch
  - effect on line printer, A–26
  - effect on terminals, A–26
  - SPOOL, A–25
- Dial-up flag
  - account attribute, 7–4
- dial-up line
  - ring characteristics, 8–5
  - set characteristics, 8–5
- /DIALUP qualifier
  - CREATE/ACCOUNT command, 7–7
  - SET ACCOUNT command, 7–14
  - SET TERMINAL command, 8–18
- Directory
  - fragmentation, 1–11
  - minimize search overhead, 1–10
  - reducing accesses, 11–2
  - restructure with REORDR, 11–2
- Directory caching, 11–7
  - cache clusters used, 11–20
  - enable on the system, 11–18
  - GFD, 11–7
  - MFD, 11–7
  - optimize, 11–11, 12–13
  - overlay code, 11–7
  - UFD, 11–7
- /DIRECTORY qualifier
  - BACKUP command, 16–6
  - RESTORE command, 16–14
- Directory structure
  - optimizing on disk, 11–2
  - prevent a damaged, 11–4
- /DIRECTORY_MAXIMUM qualifier
  - SET CACHE command, 11–20
- Dirty disk
  - rebuilding with MOUNT, 15–16
  - when dirty bit is set, 15–14
- /DISABLE qualifier
  - SET CACHE command, 11–20
  - SET DEVICE command, 11–22
  - SET NODE/LAT command, 8–12
- Disk
  - copying with IMAGE, E–13
  - create fast copy, E–1
  - differences between Level 0 and Level 1, 1–7
  - directory entries scattered, 11–2
  - dismount under SHUTUP, 4–10
  - formatting with DSKINT option of INIT.SYS, 15–2
  - free space, 14–8
  - IMAGE output as system, E–13
  - legal with IMAGE, E–13
  - MFD on system, 1–11
  - optimizing, 1–10
    - directory structure, 11–2
  - organization, 1–3
    - options, 1–3
  - private, 1–3, 1–10
  - rebuilding a dirty disk with MOUNT, 15–14
  - rebuilding with MOUNT command, 15–16
  - reorder disk structure, 11–4
  - reorder in public structure, 11–4
  - restore with SAVRES, E–1
  - restoring, E–8
- Disk (cont'd.)
  - restructure directories, 11–2
  - SAVE Set, E–1
  - statistics (DISPLY), 14–16
  - status statistics, 14–16
  - stopping an RC25, 14–2
  - system, 1–2, 1–3
  - types, 1–3
  - virtual, 11–12
- Disk characteristics, INITIALIZE display, 15–11
- Diskette controller test
  - DXEXER program, F–11
- Disk exerciser test
  - DSKEXR program, F–8
- Disk handling, DCL commands, 15–1t
- Disk quotas, 5–1
  - attribute block, 5–1
  - logged-in, 5–2
  - logged-out, 5–2
  - operations, 5–2
- Disk size, disk type, 15–8t
- Disk type, device sizes, C–2t
- DISMOUNT command
  - disks, 15–3
  - tapes, 15–5
- DISPLY
  - abbreviations in DISPLY report, 14–14
  - account location, 14–9
  - buffer status report, 12–15
  - busy device statistics, 14–16
  - commands, 14–11t to 14–12t
  - detached, 14–9
  - disk status abbreviations, 14–17t
  - disk structure statistics, 14–16
  - example, 14–13
  - free buffers, 14–18
  - information displayed, 14–10
  - INTERVAL question, 14–9
  - job status statistics, 14–14
  - memory status, 14–20
    - abbreviations, 14–21t
  - message receivers, 14–17
    - abbreviations, 14–18t
  - resident libraries, 14–20
    - abbreviations, 14–19t
  - restrict use, 14–9
  - running, 14–9
  - run-time systems, 14–18
    - abbreviations, 14–19t
  - screen layout, 14–13
  - STATE column abbreviations, 14–15t
  - stop execution, 14–10
  - switches, 14–9t
  - titles used in report, 14–14
  - use, 14–7
  - WHY column abbreviations, 14–16t
- DSKCVT program, 1–7 to 1–10
- DSKEXR program
  - disk exerciser test, F–8
  - errors, F–9
  - example dialogue, F–9
- DSKSEK program
  - data caching restriction, F–10
  - error messages, F–10
  - example dialogue, F–10
  - non-file-structured disks test, F–10
- DUMP/SYSTEM command, 12–16, 12–21

- DXEXER program
  - diskette controller test, F–11
  - error messages, F–11
  - example dialogue, F–11
- Dynamic regions, 5–4
- /DYNAMIC_REGION_LIMIT qualifier
  - SET SYSTEM command, 14–26

E

- /EIGHT_BIT qualifier
  - SET PRINTER command, 11–24
  - SET TERMINAL command, 8–18
- EIS operation test
  - CPEXER program, F–4
- Electronic messaging systems, 14–21 to 14–22
- EMT
  - definition, 14–4
  - that cannot be logged, 14–5
- EMT logging, 14–4
  - data contained in packet, 14–6
  - data returned, 14–6
  - how to program, 14–4
  - packet, 14–5
  - reasons for using, 14–4
  - shutdown under SHUTUP, 4–9
  - system security, 14–7
- /EMT_LOGGING qualifier
  - SET SYSTEM command, 14–26
  - SHOW SYSTEM command, 14–30
- /ENABLE qualifier
  - SET CACHE command, 11–21
  - SET DEVICE command, 11–22
  - SET NODE/LAT command, 8–12
- END command, OPSER, A–44
- /END qualifier
  - BACKUP command, 16–7
  - RESTORE command, 16–14
- Entry number, accessing entries, 9–7
- Entry specification, accessing entries, 9–7
- /EOT qualifier
  - SET PRINTER command, 11–24
- /ERASE qualifier
  - INITIALIZE command, 15–7
- ERR.SYS file
  - allocate and position, 12–13
- ERRCPY
  - activating, 10–1
  - active during time sharing, 10–1
  - error message from SHUTUP, 10–3
  - messages queued, 10–3
  - minimize size, 10–1
  - number of messages queued, 10–1
  - processing error messages, 10–3
  - shutdown procedures, 4–9
  - shutdown under SHUTUP, 4–9
  - total errors received, 10–10
  - use, 10–1
- ERRCRS.FIL
  - error logging file, 10–1
- ERRDET
  - criteria for bad blocks, 10–7
  - report content, 10–7
- ERRDET module
  - ERRDIS, 10–4
- ERRDIS
  - bad blocks, 10–8
- ERRDIS
  - bad blocks (cont'd.)
    - example, 10–8
    - potentially, 10–4
    - report, 10–7
  - categories of errors, 10–9
  - detailed report, 10–4
  - dialogue explanations, 10–5t
  - disk error description, 10–13t
  - error code mnemonic, 10–9
  - error file, zero contents, 10–4
  - error sequence number, 10–9
  - error title line, 10–9
  - full report, 10–9
    - examples, 10–10 to 10–19
  - functions, 10–4
  - modes, optional, 10–5
  - modules, 10–4
  - nondisk peripheral device format, 10–17, 10–18t
  - nonperipheral error
    - description, 10–19
    - format, 10–20t
  - running, 10–4
  - summary report, 10–4, 10–6
  - user description data, 10–10, 10–11t
  - zeroing the log, 10–20
- ERRINT
  - chains to ERRCPY, 10–3
  - change the maximum size, 10–2
  - command file, 10–1
  - example of dialogue, 10–3
  - location of program, 10–3
  - run, 10–1
  - run at startup, 3–7
  - validates error file, 10–1
- ERRLOG.FIL file, 10–1
  - stores error messages, 10–1
- Error file
  - adding, 12–16
  - initialization, 10–1
  - removing, 12–16
  - SHOW FILE/SYSTEM command, 12–15
  - validation, 10–1
  - zero, 10–4
- Error logging, 2–5
  - creation of file, 10–1
  - ERRCPY, 10–3
  - hardware errors, 10–1
  - initial conditions for system, 3–7
  - initialization, 10–1
  - software errors, 10–1
  - use of programs, 10–1
- Error messages
  - processing, 10–1
  - processing with ERRCPY, 10–3
  - SAVRES, E–25t
    - nonfatal transfer, E–29
  - SPOOL, A–36
    - syntax, A–28, A–29t
- Error package
  - programs in System, 10–1
- Errors
  - catastrophic, 2–2, 2–5
    - automatic recovery, 2–5
    - causes, 2–5
    - handling, 2–5
  - categories of reported (ERRCPY), 10–9

Errors (cont'd.)
- checking CCL command, 12–11
- detailed report, 10–4
- detection in RSTS/E monitor, 10–1
- disk (ERRDIS), 10–9
- displaying, 10–4
- example of MSCP in full report (ERRDIS), 10–15 to 10–17
- logging, 10–1
- missed (ERRCPY), 10–9
- MSCP in full report (ERRDIS), 10–14
- nondisk peripheral device, 10–9
- nonperipheral, 10–9, 10–19
- privileged job programming, 2–4
- recovery from line printer, A–34
- SAVRES, E–24
- SAVRES input volume, E–7t
- summary report, 10–4
- summary report (ERRDIS), 10–6
- total received by ERRCPY, 10–10

Error trap
- infinite loop, 2–5
- two categories, 2–5
- unexpected, 2–5

/ERROR_FILE qualifier
- SHOW FILE/SYSTEM command, 12–38

/ERROR_LOGGING qualifier
- INSTALL/LIBRARY command, 12–23
- INSTALL/RUNTIME_SYSTEM command, 12–26

/ESCAPE_SEQUENCE qualifier
- SET TERMINAL command, 8–18

Ethernet, 8–6

Event logger
- DECnet/E, 4–6

EVTLOG program
- DECnet/E example, 4–12
- DECnet/E shutdown, 4–6
- resetting shutdown status, 4–6
- shutdown phase, 4–8

/EXCLUDE qualifier
- BACKUP command, 16–7
- RESTORE command, 16–14

Execute access
- GREAD privilege, 6–3
- WREAD privilege, 6–7

Executive
- definition, 1–2

/EXERCISE qualifier
- INITIALIZE command, 15–8

Exit status, Batch jobs, 9–12

/EXPIRE qualifier
- CREATE/ACCOUNT command, 7–7
- SET ACCOUNT command, 7–14

EXQTA privilege
- logged-in quota, 6–3
- logged-out quota, 6–3
- Print/Batch Services (PBS), 9–5

Extended Instruction Set
- see EIS

F

FDF
- form characteristics, 9–13
- /FORMS qualifier, 9–13
- Print/Batch Services (PBS), 9–14

File
- accounting information, A–35
- add swap file, 12–13
- bad block, 10–8
- bad cluster while transferring contiguous, E–9
- caching, 11–11
  - contiguous, 11–11
  - random mode, 11–8
  - sequential mode, 11–9
- cannot remove or add NSP, 12–15
- commands
  - startup, A–13
- directory entries, 11–2
- disk structure, 1–3, 1–11
- high access (definition), 11–11
- minimize access overhead in data, 1–11
- Network Services Protocol, 12–15
- read from sequentially cached, 11–9
- remove swap file, 12–13, 12–15
- run request with no file type, 12–5
- SIL, 12–7
- structures, 1–4
- swap file, 1–12
- system control, 12–13

File access
- SYSIO privilege, 6–5

/FILE qualifier
- SET CACHE command, 11–21

/FILE_DATA qualifier
- BACKUP command, 16–7
- RESTORE command, 16–14

/FILE_END qualifier
- STOP/SERVER command, 9–47

/FILL qualifier
- SET PRINTER command, 11–24

FIP small buffer
- DISPLY report, 14–18
- use for CCL, 3–8

FMS-11, 8–9

FORCE
- Ctrl/C, 13–4
- example, 13–4

FORCE command, 13–3, 13–12

/FORMAT qualifier
- INITIALIZE command, 15–13

Formatting
- disks requiring system shutdown, 15–2

Form characteristics
- Forms Definition File (FDF), 9–13

FORM command
- SPOOL, A–36

Form length
- line printer, A–26

FORM option
- SPOOL, A–24

Forms alignment
- changing, A–36
- during time sharing, A–25
- procedures, A–25

Forms control, A–25

Forms Management System
- See FMS-11

/FORMS qualifier
- Forms Definition File (FDF), 9–13
- INITIALIZE/QUEUE command, 9–26
- INITIALIZE/SERVER command, 9–29
- SET QUEUE command, 9–33

/FORMS qualifier (cont'd.)
  SET SERVER command, 9–35
/FORM_FEED qualifier
  SET TERMINAL command, 8–18
/FORWARDSPACE qualifier
  START/QUEUE/MANAGER command, 9–42
FPP test
  CPEXER program, F–4
/FULL qualifier
  BACKUP command, 16–7
  RESTORE command, 16–14
  SHOW ACCOUNT command, 7–20
  SHOW SERVER command, 9–38
  SHOW TERMINAL command, 8–30

G

GACNT privilege
  account management, 6–3, 6–7
General small buffer
  DISPLY report, 14–18
  missed errors, 10–10
  SPOOL error, A–29
  use for CCL, 3–8
GFD (Group File Directory)
  with MFD and UFDs in Level 1, 1–6f
GREAD privilege
  execute access, 6–3
  read access, 6–3
/GROUP_SIZE qualifier
  BACKUP command, 16–7
GWRITE privilege
  create access, 6–3
  rename access, 6–3
  write access, 6–3

H

HANGUP command, 13–7, 13–13
/HANGUP qualifier
  SET SYSTEM command, 14–26
/HARDCOPY qualifier
  SET TERMINAL command, 8–18
Hardware
  malfunctions, 2–4
Hardware controller
  terminal device designator, 8–2
Hardware errors
  logging, 10–1
/HEADER qualifier
  BROADCAST command, 13–10
Heading burst pages, A–25
HELLO command, CCL, 3–8
Help report
  ERRDIS (example), 10–6
/HOLD qualifier
  SET JOB command, 13–17
  SET SYSTEM command, 14–26
HOLD state
  processing entries, 9–6
/HOSTSYNC qualifier
  SET TERMINAL command, 8–18
HWCFG privilege
  SET TERMINAL command, 8–3
  system manager function, 6–3
  terminal characteristics, 8–3
HWCTL privilege
  system control, 6–4

I

/IDENTIFICATION qualifier
  CREATE/SERVICE/LAT command, 8–9
  SET NODE/LAT command, 8–12
  SET SERVICE/LAT command, 8–13
IDENTIFY, E–3t
  dialogue, E–17t
  example, E–17
IMAGE, E–3t
  dialogue, E–13t
  example, E–15 to E–16
  extracting pack ID, E–17t
  output from, as system disk, E–13
  run statistics, E–23
  SAVRES, E–13
Immediate mode
  shutting down OPSER, 4–8
/INCLUDE qualifier
  BACKUP command, 16–7
  RESTORE command, 16–14
Indexed file
  caching, 11–11
/INDEX qualifier
  INITIALIZE command, 15–9
INIT
  command, 2–5
  start OPSER, A–13
INIT.BAS program
  OPSER shutdown conditions, 4–8
INIT.SYS file
  copies to disk, E–18
INIT.SYS program
  control file, 3–1
  DSKINT option to format disks, 15–2
  use, 2–1
Initialization code
  bootstrapped, 2–1
  RSTS/E, 2–2
  START option, 3–1
Initialization options
  discussion, 1–2
  requesting, 2–3
  summary, 2–3t
INITIALIZE/QUEUE command, 9–25
INITIALIZE/SERVER command, 9–29
INITIALIZE command
  /DENSITY qualifier, 15–12
  disks, 15–6 to 15–11
  display of disk characteristics, 15–11
  /FORMAT qualifier, 15–13
  QUEMAN, A–15
  tapes, 15–12
/INITIALIZE qualifier
  BACKUP command, 16–8
  RESTORE command, 16–14
Initializing disks
  INITIALIZE command, 15–6 to 15–11
  overview, 15–1
Initializing tapes
  INITIALIZE command, 15–12
  overview, 15–1
/INQUIRE qualifier
  SET TERMINAL command, 8–19

INSTALL/ERROR_FILE command, 12–13, 12–16, 12–22
INSTALL/LIBRARY
  DCL command, 12–7, 12–8
  /LOCK qualifier, 12–8
  /[NO]ADDRESS[=n] qualifier, 12–7
INSTALL/LIBRARY command, 12–23
INSTALL/OVERLAY_FILE command, 12–13, 12–16, 12–25
INSTALL/RUNTIME_SYSTEM
  /ADDRESS=n qualifier, 12–4
  /LOCK qualifier, 12–6
  /MAXIMUM=s qualifier, 12–6
  /MINIMUM=s qualifier, 12–6
  /[NO]ERROR_LOGGING qualifier, 12–4
  /[NO]KEYBOARD_MONITOR qualifier, 12–4
  /[NO]READ_ONLY qualifier, 12–4
  /[NO]SHAREABLE qualifier, 12–4
  /[NO]UNLOAD qualifier, 12–4
  /POSITION qualifier, 12–5
  /PREFIX=n qualifier, 12–4
INSTALL/RUNTIME_SYSTEM command, 3–7, 12–4, 12–26
  errors, 12–5
INSTALL/RUNTIME_SYSYTEM
  /TYPE qualifier, 12–6
INSTALL/SWAP_FILE
  errors, 12–15
  /SIZE=n qualifier, 12–14
INSTALL/SWAP_FILE command, 12–13, 12–14, 12–28, 13–2
INSTAL privilege
  defining CCLs, 12–10
  system commands, 6–4
  system files, 6–4
  system logical names, 6–4
/INTERACTIVE qualifier
  CREATE/ACCOUNT command, 7–7
  SET ACCOUNT command, 7–14
  SET TERMINAL command, 8–19
INTERRUPT command
  OPSER, A–11, A–12, A–17t, A–44
  response to, A–12
Interrupt commands
  BATCH, A–40t
  SPOOL, A–30
/IN_LABEL qualifier
  BACKUP command, 16–8
  RESTORE command, 16–14
/IN_QUOTA qualifier
  CREATE/ACCOUNT command, 7–7
  SET ACCOUNT command, 7–14

J

Job
  accounting information, A–35
  check detached, 14–8
  check number of, 14–8
  compute bound, 13–6
  definition, 1–1
  executing a read request, 11–7
  header, A–35
  keyboard monitor, 1–1
  limit of spooling, A–20
  maximum (JOB MAX), 12–14
  maximum assigned size, 13–7
Job (cont'd.)
  number allowed, 12–14
  OPSER checks online, A–5
  OPSER table, A–2
  priority, 13–6
  run burst, A–32
  set maximum size, 13–5
  setting priority, 13–5
  size, 2–6
  status statistics, 14–14
  suspend, 13–5
Job area
  size, 1–11
  user, 1–11
Job control
  SHOW USER command, 13–4
JOBCTL privilege
  system control, 6–4
JOB MAX
  restricts jobs, 12–15
/JOB qualifier
  ALLOCATE command, 11–13
Job quota, 5–3
  detached-job, 5–3
  log-in request, 5–3
Job quotas, 5–1
Job run burst, 13–5
Job status, SYSTAT, 14–8
Job table, SHUTUP, 4–6
/JOB_END qualifier
  STOP/SERVER command, 9–47
/JOB_QUOTA qualifier
  CREATE/ACCOUNT command, 7–7
  SET ACCOUNT command, 7–14

K

/KATAKANA qualifier
  SET TERMINAL command, 8–19
KBEXER program
  errors, explanation, F–13
  example dialogue, F–12
  local terminals test, F–12
  remote terminals test, F–12
  terminal tests, F–12
/KEBOARD_MONITOR qualifier
  SET JOB command, 13–17
/KEEP=n qualifier
  minimum residency time, 11–11
/KEEP qualifier
  SET CACHE command, 11–21
Keyboard
  controlling, 13–7
  default SPOOL values, A–28
  delimiter, 13–6
  monitor, 1–11
  spooling, A–28
  startup on LA180, A–33
  startup on LA36, A–32
Keyboard monitor, 3–8
  change with SWITCH, 1–1
  default (definition), 1–1
  definition, 1–2
  job, 1–1
  process command, 1–11
/KEYBOARD_MONITOR qualifier
  INSTALL/RUNTIME_SYSTEM command, 12–27

KILL
UTILTY program, A–45

L

LA180
form length, A–26
startup keyboard on, A–33
LA36
startup keyboard on, A–32
Label, specified for tapes
DISMOUNT command, 15–5
INITIALIZE command, 15–12
MOUNT command, 15–19
/LABEL qualifier
SET SYSTEM command, 14–27
LAT, 8–6 to 8–8
enabling at startup, 3–9
enabling LAT, 8–7
LAT problems, 8–8
logins, 13–2
/LAT qualifier
SET SYSTEM command, 8–14, 14–27
SHOW SESSIONS/LAT command, 8–28
Level 0, 1–5f
compared to Level 1, 1–7
Level 1
compared to Level 0, 1–7
three structures involved, 1–6
Library
remove resident, 12–8
resident, 12–7
LIBRARY keyword, 12–7
Line printer
default SPOOL values, A–27
/DFLENGTH switch effect, A–26
distinction from terminal, A–27
form length, A–26
output, A–35
recovery from errors, A–34
requests for output, A–4
spooling, A–21, A–27
startup
all defaults, A–31
narrow width, A–31
Line printer test
LPEXER program, F–13
/LINE_NUMBER qualifier
DEFINE/COMMAND/SYSTEM command, 12–19
LIST command
OPSER, A–12
LIST JOBS command
OPSER, A–12, A–43
LIST OPERATORS command
OPSER, A–12
/LIST_FILE qualifier
BACKUP command, 16–8
RESTORE command, 16–15
LOAD/INDEX command, 14–22
LOAD/OVERLAY command, 14–23
/LOADABLE_CHARACTERS qualifier
SET TERMINAL command, 8–19
Load address
run-time system, 12–4
Local Area Transport
See LAT
Local terminal test
KBEXER program, F–12
/LOCAL_ECHO qualifier
SET TERMINAL command, 8–19
/LOCK qualifier
INSTALL/LIBRARY command, 12–23
INSTALL/RUNTIME_SYSTEM command, 12–27
Log file
closing
under SHUTUP, 4–10
LOGFILE command
OPSER, A–13
Logged-in quota, 5–2
EXQTA privilege, 6–3
Logged-out quota, 5–2
EXQTA privilege, 6–3
/LOGGED_IN qualifier
BROADCAST command, 13–10
FORCE command, 13–12
Logical device
BATCH, A–38
SPOOL, A–22
Logical end mode
shutdown of OPSER, 4–8
Logical name
account number associated, 12–8
adding new, 12–9
ASSIGN/SYSTEM command, 12–9
DEASSIGN/SYSTEM command, 12–10
job related, 12–8
list, 12–10
system, 12–8
system defined, 12–8
unique, 12–9
Logical Names
managing with DCL commands, 12–1t
LOGIN
set priority, 13–6
LOGIN.COM
LAT reporting, 8–8
login command file, 6–8
LOGIN command, 13–14
LAT reporting, 8–8
Login command file
LOGIN.COM, 6–8
Logins
decrease the number of, 12–15
limit, 13–2
limit for DECnet/E, 4–4
limiting, in SHUTUP, 4–4
limit under SHUTUP, 4–4
over LAT lines, 13–2
restrict, 13–1
restricted by swap space, 12–15
swap file space affects, 12–15
SWCTL privilege, 6–5
/LOGINS qualifier
SET SYSTEM command, 13–19, 14–27
LOGOUT command, 13–14
LAT reporting, 8–8
/LOG qualifier
CREATE/ACCOUNT command, 7–7
CREATE/SERVICE/LAT command, 8–10
CREATE/VIRTUAL_DISK command, 11–15
DELETE/ACCOUNT command, 7–10
DELETE/SERVICE/LAT command, 8–11
DELETE/VIRTUAL_DISKcommand, 11–17

/LOG qualifier (cont'd.)
  INSTALL/ERROR_FILE command, 12–22
  INSTALL/OVERLAY_FILE command, 12–25
  INSTALL/SWAP_FILE command, 12–28
  SET ACCOUNT command, 7–14
  SET FILE command, 12–36
  SET NODE/LAT command, 8–12
  SET PASSWORD command, 7–17
  SET SERVICE/LAT command, 8–13
  SET SYSTEM command, 14–27
  START/LAT command, 8–32
  STOP/LAT command, 8–33
/LOOKUP qualifier
  CREATE/ACCOUNT command, 7–7
  SET ACCOUNT command, 7–14
/LOWERCASE qualifier
  SET PRINTER command, 11–24
  SET TERMINAL command, 8–19
LPEXER program
  errors, explanation, F–13
  example dialogue, F–13
  line printer test, F–13

M

Magnetic tape
  density for Save Set, E–20
  /DENSITY switch, E–6
  SAVE Set, E–1, E–19
Magnetic tape exerciser
  See MTEXER program
MAKSIL program
  formats disk file, 12–7
Management
  RSTS/E system, 1–2
/MAXIMUM qualifier
  SET CACHE command, 11–21
/MAXIMUM_SIZE qualifier
  INSTALL/RUNTIME_SYSTEM command, 12–27
Memory
  adding resident library, 12–8
  available range of, 12–5
  avoiding fragmentation, 12–4
  dump, 10–21
  fragmentation, 12–5
  high, 12–5
  low, 12–5
  status statistics, 14–20
Memory access
  data caching, 11–7
Memory dump
  annotated version, 10–22
  critical contents, 10–23
MESSAGE command
  OPSER, A–10, A–13
Message quota, 5–4
Message receiver
  OPSER, A–2
  QUEMAN declares self, A–15
Message receivers
  abbreviations (DISPLY), 14–18
  statistics (DISPLY), 14–17
Messages
  ? characters, D–1
  % characters, D–1
  general, D–1, D–2t to D–29t
  informational, D–1
Messages (cont'd.)
  severe error, D–1
  standard error, D–1
  warning, D–1
/MESSAGE_QUOTA qualifier
  CREATE/ACCOUNT command, 7–8
  SET ACCOUNT command, 7–14
Messaging systems, 14–21 to 14–22
MFD (Master File Directory)
  and UFDs in Level 0, 1–5f
  with GFDs and UFDs in Level 1, 1–6f
/MFD_CLUSTER_SIZE qualifier
  INITIALIZE command, 15–9
/MINIMUM_SIZE qualifier
  INSTALL/RUNTIME_SYSTEM command, 12–27
Missed errors
  reported to ERRCPY, 10–9
MODE values
  data caching, 11–7
/MODIFIED qualifier
  BACKUP command, 16–5
  RESTORE command, 16–13
Monitor
  caching checks, 11–19f
  changing monitors, 2–6
  control of run-time system, 12–3
  customizing, 2–6
  default keyboard, 1–1
  definition, 1–2
  directives (data caching), 11–7
  error detection, 10–1
  job keyboard, 1–1
  keyboard, 1–2
  symbols (list), 10–23
  take online dump, 12–16
  update cache, 11–8
/MONITOR_NAME qualifier
  SET SYSTEM command, 14–27
MOUNT command (disks), 15–14
  /[NO]QUOTA qualifier, 15–16
  /[NO]REBUILD qualifier, 15–16
  /[NO]RESTRICT qualifier, 15–17
  /[NO]SHARE qualifier, 15–16t, 15–18
  /[NO]WRITE qualifier, 15–18
  /OVERRIDE qualifier, 15–15
  /PRIVATE qualifier, 15–15, 15–16t
  /PUBLIC qualifier, 15–15, 15–16t
MOUNT command (tapes), 15–19
  /DENSITY qualifier, 15–19
  /FORMAT qualifier, 15–20
  /OVERRIDE qualifier, 15–20
  /WRITE qualifier, 15–20
Mounting disks
  DISMOUNT (disks), 15–3
  MOUNT command (disks), 15–14 to 15–19
  overview, 15–2
Mounting tapes
  DISMOUNT (tapes), 15–5
  MOUNT (tapes), 15–19
  overview, 15–2
MOUNT privilege
  public disk, 15–14
  shared disks, 6–4
MSCP
  ERRDIS full report, 10–14
    example, 10–15 to 10–17

MTEXER program
- error messages, F–14
- example dialogue, F–14
- magnetic tape controllers test, F–13
- magnetic tape drives test, F–13

/MULTICAST_TIMER qualifier
- SET NODE/LAT command, 8–12

N

NAME option
- SPOOL, A–24

/NAME qualifier
- CREATE/ACCOUNT command, 7–8
- SET ACCOUNT command, 7–14
- SET SYSTEM command, 14–27

NCP
- network shutdown, 4–6

Network
- shutdown of DECnet/E, 4–5

Network management
- SWCTL privilege, 6–5

/NETWORK qualifier
- CREATE/ACCOUNT command, 7–8
- SET ACCOUNT command, 7–14

/NETWORK_FILE qualifier
- SHOW FILE/SYSTEM command, 12–38

/NEW_FILES qualifier
- INITIALIZE command, 15–10

NEXT command
- error in, A–19
- error text, A–19t
- QUEMAN, A–19

/NEXT_JOB qualifier
- START/QUEUE/MANAGER command, 9–43

/NOBROADCAST qualifier, prevents receiving broadcasts, 8–6

/NOCONTIGUOUS qualifier
- SET FILE command, 12–35

/NODE qualifier
- SHOW COUNTERS/LAT command, 8–24

Non-file-structured I/O
- RDNFS privilege, 6–4
- WRTNFS privilege, 6–7

Non-file-structure disk test
- DSKSEK program, F–10

Nonuser account, 7–1

NORMAL option
- changing in SPOOL, A–24
- SPOOL, A–24

/[NO]FMS qualifier
- SET SYSTEM command, 8–14, 14–26

/[NO]STATISTICS qualifier
- SET SYSTEM command, 14–28

NSP (Network Services Protocol)
- cannot remove or add, 12–15

Number conversion, bit position, B–1

O

/OCCURRENCE qualifier
- RESTORE command, 16–15

ODT (Octal Debugging Tool)
- setting breakpoints with, 12–4

/OMIT_CR qualifier
- SET PRINTER command, 11–24

ONLCLN
- perform disk cleaning, E–7

OPEN/QUEUE command, 9–31

Operating system
- definition, 1–2

OPERATOR command
- OPSER, A–13

Operator communication program, A–42

Operator services
- BATCH, A–1
- flowchart, A–3f
- OPSER, A–1, A–6
- overview, A–1
- QUEMAN, A–1
- RESTOR, A–1
- SPOOL, A–1

Operator table
- OPSER, A–6

OPSER
- ABORT command, A–45
- action requests, A–10
  - contents, A–11t
- ANSWER command, A–11
- cause INIT to start, A–13
- CHANGECONSOLE command, A–13
- changes OSC, A–13
- checks for on line jobs, A–4, A–6
- checks operator table, A–6
- commands
  - from PLEASE, A–43t
  - from SHUTUP, A–44
  - START.COM, A–13
- communicates
  - three ways, A–4
  - with BATCH, A–40
- communicates with SHUTUP, 4–2
- controlled program, A–2
- controlling RESTOR, A–5, A–46
- DETACH command, A–13
- END command, A–44
- immediate shutdown mode, 4–8
- initial operating conditions for, A–13
- interaction with action request, A–4
- interaction with information line, A–4
- interaction with message, A–4
- INTERRUPT command, A–11, A–12, A–17t, A–44
- LIST command, A–12
- LIST JOBS command, A–12, A–43
- LIST OPERATORS command, A–12
- LOGFILE command, A–13
- logical end mode, 4–8
- message
  - contents, A–11t
  - formats, A–10
  - levels, A–11
  - receiver identification, A–2
- MESSAGE command, A–10, A–13
- online job list, A–12t
- OPERATOR command, A–13
- operator commands, A–8t
- operator communicate with, A–43
- operator table, A–6
- overview, A–2
- RESTOR commands, A–46t
- restrictions of, A–2
- RETYPE command, A–10
- running with SHUTUP, 4–3

OPSER (cont'd.)
  send command to QUEMAN, A–18
  shutdown, A–5
  shutdown in SHUTUP, 4–8
  shutdown procedures, 4–8
  start after system crash, A–14
  starting, A–6
  startup procedure, A–13
  table of online jobs, A–1
  terminating with SHUTUP, A–5, A–44
  valid operators, A–12, A–43
  ways terminated, A–13
  work file, A–6
OPSER.LOG file
  default log file, A–7
OPSER package
  BATCH, A–37
OPSER Package Library
  QUEMAN, A–14
/OPTIMIZE qualifier
  RESTORE command, 16–15
Options
  INIT program, 2–5
OSC (Operator Services Console)
  INTERRUPT responses, A–12
  OPSER broadcasts messages to, A–2
  OPSER changes, A–13
  output formats, A–10
  send text to, A–42
/OUTPUT qualifier
  BACKUP command, 16–9
  RESTORE command, 16–16
  SHOW ACCOUNT command, 7–20
  SHOW COUNTERS/LAT command, 8–24
  SHOW NODE/LAT command, 8–25
  SHOW SERVICE/LAT command, 8–26
  SHOW SESSIONS/LAT command, 8–28
  SHOW TERMINAL_SERVER/LAT command, 8–31
/OUT_LABEL qualifier
  BACKUP command, 16–8
  RESTORE command, 16–16
/OUT_QUOTA qualifier
  CREATE/ACCOUNT command, 7–8
OUT_QUOTA qualifier
  SET ACCOUNT command, 7–14
Overlay file
  adding, 12–16
  removing, 12–16
  SHOW FILE/SYSTEM command, 12–15
/OVERLAY_FILE qualifier
  SHOW FILE/SYSTEM command, 12–38
/OVERRIDE qualifier
  LOGIN command, 13–14
OVR.SYS file
  allocate and position, 12–13

P

Pack identification
  extracting with SAVRES, E–17t
/PAGE qualifier
  START/QUEUE/MANAGER command, 9–43
/PAGE_EJECT switch
  SPOOL, A–25
/PAGE_END qualifier
  STOP/SERVER command, 9–47
/PAGE_LENGTH qualifier
  SET PRINTER command, 11–24
/PAGE_LIMIT qualifier
  INITIALIZE/QUEUE command, 9–27
  SET QUEUE command, 9–33
/PARITY qualifier
  SET TERMINAL command, 8–19
/PASSWORD_PROMPT qualifier
  CREATE/ACCOUNT command, 7–8
  SET ACCOUNT command, 7–15
  SET SYSTEM command, 14–27
PBS
  batch servers, 9–2
  default queues, 9–3
  Forms Definition File (FDF), 9–14
  print servers, 9–2, 9–7
  queue manager, 9–2, 9–3
  queue names, 9–3
  queue priorities, 9–5
  queue types, 9–3
  treatment by SHUTUP, 4–6
  user interface, 9–2
PBSCTL privilege
  Print/Batch Services (PBS), 6–4
PDP-11 word
  decimal values, B–1
  octal values, B–1
PEEK function test
  CPEXER program, F–4
/PERMANENT qualifier
  SET TERMINAL command, 8–5, 8–20
  SHOW TERMINAL command, 8–30
Physical device names, 12–8
PHYSICAL option
  SPOOL, A–24
/PLACED qualifier
  SET FILE command, 12–36
/PLACED_POSITION qualifier
  RESTORE command, 16–16
.PLAS system directive
  RDMEM privilege, 6–4
PLEASE
  CCL command, A–44
  commands through OPSER, A–43t
  commands to OPSER, A–43
  message broadcast, A–43
  OPSER commands through, A–43
  restrict use, A–42
  run, A–42
  talks to OPSER, A–3
  terminate, A–42
/POSITION qualifier
  CREATE/ACCOUNT command, 7–8
  INSTALL/ERROR_FILE command, 12–22
  INSTALL/OVERLAY_FILE command, 12–25
  INSTALL/RUNTIME_SYSTEM command, 12–27
  INSTALL/SWAP_FILE command, 12–29
Power fail
  automatic restart after, 2–6
/POWERFAIL_DELAY qualifier
  SET SYSTEM command, 14–27
/PREFIX qualifier
  INSTALL/RUNTIME_SYSTEM command, 12–27
Primary run-time system, 12–3
  position, 12–5
  system disk, 1–3

- Print/Batch Service (PBS)
  - ASSIGN/QUEUE command, 9–20
  - INITIALIZE/QUEUE command, 9–25
- Print/Batch Services
  - PBSCTL privilege, 6–4
- Print/Batch services (PBS)
  - STOP/SERVER command, 9–46
- Print/Batch Services (PBS)
  - CLOSE/QUEUE command, 9–21
  - DEASSIGN/QUEUE command, 9–22
  - DELETE/QUEUE command, 9–23
  - DELETE/SERVER command, 9–24
  - INITIALIZE/SERVER command, 9–29
  - many-to-one configuration, 9–4
  - one-to-many configuration, 9–4
  - one-to-one configuration, 9–4f
  - OPEN/QUEUE command, 9–31
  - SET QUEUE command, 9–32
  - SET SERVER command, 9–35
  - START/QUEUE/MANAGER command, 9–40
  - START/QUEUE command, 9–39
  - START/SERVER command, 9–42
  - STOP/QUEUE/MANAGER command, 9–45
  - STOP/QUEUE command, 9–44
- Print/Batch Services Package (PBS)
  - DCL commands, 9–1t
  - managing, 9–2 to 9–12
- Printer servers
  - ownership, 9–8
- /PRINTER_PORT qualifier
  - SET TERMINAL command, 8–20
- Printing device
  - adjustable top of form, A–26
- /PRINT qualifier
  - INITIALIZE/QUEUE command, 9–27
  - SET QUEUE command, 9–33
  - SET SERVER command, 9–36
  - SHOW SERVER command, 9–38
- Print servers
  - Print/Batch Services (PBS), 9–2, 9–7
  - wildcard names, 9–8
- Priority
  - basis, 13–5
  - byte, 13–6
    - format, 13–5f
  - decrease for job, 13–6
  - description, 13–5
  - LOGIN, 13–6
  - processing entries, 9–6
  - raise for SPOOL, A–32
  - setting job, 13–5
- /PRIORITY qualifier
  - INITIALIZE/QUEUE command, 9–27
  - INITIALIZE/SERVER command, 9–30
  - SET JOB command, 13–17
  - SET QUEUE command, 9–34
  - SET SERVER command, 9–36
  - START/QUEUE/MANAGER command, 9–41
- /PRIORITY switch
  - QUEMAN, A–16t
- Private disk, 1–3, 1–10
- /PRIVATE qualifier
  - INITIALIZE command, 15–10
- Privilege
  - controlled access, 1–12
  - system damage, 1–12
- Privilege bit
  - TMPPRV privilege, 6–6
- Privileged job
  - programming error, 2–4
- Privilege mask
  - authorized, 6–8
  - current, 6–8
  - saved, 6–8
- /PRIVILEGE qualifier
  - CREATE/ACCOUNT command, 7–8
  - SET ACCOUNT command, 7–15
  - SET JOB command, 13–17
- /PRIVILEGES qualifier
  - DEFINE/COMMAND/SYSTEM command, 12–19
  - INITIALIZE/QUEUE command, 9–27
  - SET QUEUE command, 9–34
- Processing entries
  - AFTER state, 9–6
  - HOLD state, 9–6
  - priority, 9–6
  - READY state, 9–6
- Program
  - system, 1–2
- Program development, 1–2
- Programming errors
  - causing crash, 2–5
- /PROMPT qualifier
  - BACKUP command, 16–9
  - RESTORE command, 16–16
- /PROTECTION qualifier
  - INSTALL/LIBRARY command, 12–23
  - SET FILE command, 12–36
- Pseudo keyboard
  - BATCH, A–38
  - use with BATCH, A–5
- Pseudo keyboards, 8–8
  - batch servers, 9–12
- /PSEUDO_KEYBOARDS qualifier
  - SET SYSTEM command, 8–14, 14–27
- Public disk structure, 1–3
- /PUBLIC qualifier
  - DISMOUNT command, 15–3
  - INITIALIZE command, 15–10
- Public structure, 1–2
- /PURGE qualifier
  - DELETE/ACCOUNT command, 7–10

Q

- QUE, A–4
- QUE command
  - SPOOL, A–35
- QUEMAN
  - command received through OPSER, A–18
  - commands
    - startup, A–19
  - consistency checks, A–15, A–20
  - DETACH command, A–16
  - do not Ctrl/C, A–16
  - entry in AFTER queue, A–20
  - error in NEXT command, A–19
  - initial conditions for spooling queues, A–15
  - INITIALIZE command, A–16
  - integrity checks, A–15
  - message receiver declaration, A–15
  - modules, A–14
  - OPSER Package Library, A–14

QUEMAN (cont'd.)
  requirements for running, A–14
  resets to initial states, A–19
  running
    after SHUTUP, A–20
    after system crash, A–20
  running detached, A–16, A–19
  start, A–14
  startup
    command file, A–19
    commands, A–16t
    switches, A–16t
  STATUS command, A–18
  status printout, A–18
  table of online spooling programs, A–4
/QUERY qualifier
  BACKUP command, 16–9
  DELETE/ACCOUNT command, 7–11
  INITIALIZE command, 15–11
  REMOVE/JOB command, 13–16
  RESTORE command, 16–16
QUEUE.SYS file, A–4
  clear all entries from, A–15
  consistency checking on, A–20
  corrupt, A–16
  flag value, A–15
  initial conditions, A–15
  number of queue requests, A–20
Queue limits, 9–7
Queue management, A–14
Queue manager
  Print/Batch Services (PBS), 9–2, 9–3
Queue names
  Print/Batch Services (PBS), 9–3
Queue priorities
  Print/Batch Services (PBS), 9–5
Queues
  starting, 9–5
  stopping, 9–5
Queue types
  Print/Batch Services (PBS), 9–3
Quotas
  Disk, 5–1
  dynamic regions, 5–4
  Job, 5–1
  message, 5–4
  RIB (Receiver ID Block), 5–4
  send/receive, 5–1

R

Random data caching, 11–8
RC25
  stopping, 14–2
RDMEM privilege
  memory mapping, 6–4
  read memory, 6–4
RDNFS privilege
  non-file-structured I/O, 6–4
RDS (RSTS/E Disk Structure)
  levels, 1–4
  SYSTAT, 1–4
Read access
  GREAD privilege, 6–3
  WREAD privilege, 6–7
READY state
  processing entries, 9–6
/READ_ONLY qualifier
  INSTALL/LIBRARY command, 12–24
  INSTALL/RUNTIME_SYSTEM command, 12–27
Rebuilding dirty disks with MOUNT, 15–16
Receiver identification, A–24
RECOVR.COM file, 16–2
/REGIS qualifier
  SET TERMINAL command, 8–20
/RELEASE qualifier
  SET JOB command, 13–18
  SET SYSTEM command, 14–27
Remote line, controlling, 13–7
Remote terminal test
  KBEXER program, F–12
REMOVE/ERROR_FILE command, 12–16, 12–30
REMOVE/JOB command, 13–16
  example, 13–4
REMOVE/LIBRARY command, 12–31
REMOVE/OVERLAY_FILE command, 12–16, 12–32
REMOVE/RUNTIME_SYSTEM command, 12–6, 12–33
REMOVE/SWAP_FILE command, 12–15, 12–34
Rename access
  GWRITE privilege, 6–3
  WWRITE privilege, 6–7
REORDR
  account designators, 11–4
  dialogue, 11–3
    questions, 11–3t
  directory restructuring functions, 11–2
  error messages, 11–5
  example, 11–6
  fatal errors, 11–6
  messages, 11–6t
  precautions on use, 11–5
  processing messages, 11–5
  restructure directories, 11–2
  run, 11–3
/REPLACE qualifier
  ASSIGN/SYSTEM command, 12–17
  RESTORE command, 16–2, 16–17
REQUE command
  SPOOL, A–34
/RESET qualifier
  DELETE/ACCOUNT command, 7–11
  SET TERMINAL command, 8–20
  SHOW ACCOUNT command, 7–20
Resident libraries
  DISPLY program status, 14–20
Resident library, 12–7
  memory location, 12–7
  permanently resident, 12–8
  remove, 12–8
  removing
    under SHUTUP, 4–9
  RMS, 11–10
  XBUF allocation, 11–10
RESTART command
  SPOOL, A–34
/RESTART qualifier
  START/QUEUE/MANAGER command, 9–43
RESTOR
  attached to a terminal, A–46
  commands through OPSER, A–46t
  running detached, A–5, A–46
RESTORE, E–3t
  aborting, E–21

RESTORE (cont'd.)
  bad blocks, E–8
  dialogue questions, E–10t
  example, E–11 to E–13
  output used for system disk, E–10
  relocation of blocks, E–9
  restoring a RSTS/E disk, E–8
  run statistics, E–23
  SAVE Set format, E–1
RESTORE command, 16–11 to 16–18
  quick reference table, 16–18t
  restrictions, 16–11
  system recovery, 16–2
Restricted device
  DEVICE privilege, 6–3
Restricted receiver
  SEND privilege, 6–5
Restrictions
  Backup sets, 16–4
  RESTORE command, 16–11
/RESTRICT qualifier
  SET DEVICE command, 11–22
  SET TERMINAL command, 8–20
/RESUME qualifier
  SET TERMINAL command, 8–20
/RETAIN qualifier
  INITIALIZE command, 15–11
Retrieval pointers
  information for data blocks, 11–2
RETYPE command, OPSER, A–10
/REWIND qualifier
  BACKUP command, 16–9
  RESTORE command, 16–17
RIB (Receiver ID Block), 6–4
  quota, 5–4
/RIB_QUOTA qualifier
  CREATE/ACCOUNT command, 7–8
  SET ACCOUNT command, 7–15
Ring characteristics, 8–1
RJ2780 program
  communication paths, A–5
  shutdown, A–6
RMS (Record Management System)
  caching files, 11–11
  indexed file bucket sizes, 11–11
  resident library, 11–10
RSTS/E
  bootstrapping after system halt, 2–2
  bootstrapping procedures, 2–1
  disk structures, 1–3
  file structure, 1–4
  halting, 2–1
  initialization code, 2–1
  initialization options, 2–1
  management, 1–2
  monitor error detection, 10–1
  run-time systems, 12–3
  shutdown procedures, 4–1
  starting up, 2–1
RSTS/E Error Package
  use with DEVTST package, F–1
RSTS/E Privileges, 6–2t
RSX, keyboard monitor, 1–2
RT11, keyboard monitor, 1–2
Run burst
  decrease, A–32
  default value, 13–6

Run burst (cont'd.)
  definition, 13–6
  raise on SPOOL, A–32
  set, 13–5
  system operation, 13–6
/RUNBURST switch
  QUEMAN, A–16t
RUN command
  how system treats, 12–5
  no file type specified, 12–5
Run-time system, 1–2
  auxiliary, 12–3, 12–4
  BASIC-PLUS, 12–3
  change default file type, 12–6
  control, 12–3, 12–4
  DCL command, 12–4
  definition, 12–3
  description block, 12–4
  DISPLY report, 14–18
  installing, 3–7
  language interface, 12–3
  loading a, 12–5
  permanently resident, 12–6
  placement of, 12–5
  position of primary, 12–5
  primary, 12–3
  removing
    auxiliary, 12–4
    from system, 12–6
    under SHUTUP, 4–9
  .RTS file type, 12–3
  specifying load address, 12–4
  storage of, 12–3
  temporarily resident, 12–6
  use, 12–3
/RUNTIME_SYSTEM qualifier
  SET FILE command, 12–36
/RUN_BURST qualifier
  INITIALIZE/SERVER command, 9–30
  SET JOB command, 13–18
  SET SERVER command, 9–36
  START/QUEUE/MANAGER command, 9–41

S

SAV/RES
  full function command line, E–18
SAV/RES FUNCTION
  prompt, E–2, E–3
SAVE
  run statistics, E–23
SAVE Set
  default name, E–2
  dismount message, E–20
  extract label information from, E–17t
  format, E–1, E–8
  identify, E–2
  magnetic tape, E–19
  magnetic tape density, E–20
  processing multi-volume, E–20
  recreate disk, E–8
  volume
    booting RSTS/E, E–19t
    by-pass label checking, E–6
    SAVRES, E–1
SAVRES
  bad blocks

SAVRES
  bad blocks (cont'd.)
    output device, E–24
    processing, E–24
  changes caused by bad blocks, E–24
  checking
    input volume, E–7
  comparison with BACKUP, E–1
  create bootable medium, E–18
  Ctrl/C in dialogue, E–5, E–21
  Ctrl/Z in dialogue, E–5
  definition of terms, E–1
  device specification switches, E–6t
  dialogue, E–3
    defaults, E–5
  disks, like, E–2
  dismounting volumes, E–20
  dismount message, E–20
  errors, E–3
    dialogue, E–24
    handling, E–24
    messages, E–25t
    transfer phase, E–24
    verification phase, E–24
  /ERROR switch, E–4, E–5t
  exit, E–3
  fatal transfer errors, E–28
  full function command line, E–4
  function response, E–3
  functions, E–1, E–3t
  IDENTIFY
    dialogue, E–17t
    example, E–17
  IMAGE
    dialogue, E–13t
    example, E–15 to E–16
    function, E–13
  input volume errors, E–7t
  message types, E–3
  mounting volumes, E–20
  /NOERROR switch, E–5t, E–24, E–28
  nonfatal transfer errors, E–28, E–29t
  nonselective backup, E–1
  /NOSTATS switch, E–5t
  /NOVERIFY switch, E–4, E–5t
  operator intervention during processing, E–19
  prompt, E–3
  reaccessing devices, E–21
  RESTORE
    dialogue questions, E–10
    example, E–11 to E–13
    output as system disk, E–10
    processing, E–2
  restore archived sets, E–1
  RESTORE function, E–8, E–9
  run, E–2
    off line, E–2
    on line, E–2
  SAVE set, E–1
  SAVE Set
    name, E–2
    volume, E–1
  /STATS, E–5t
  /STATS switch, E–4
  summary information, E–4
  summary report, E–22
    format, E–22f

SAVRES
  summary report (cont'd.)
    totals, E–23t
  switch types, E–4
  transfer errors, E–28
  use, E–1
  /VERIFY switch, E–4, E–5t
  volume label checking, E–18
/SCOPE qualifier
  SET TERMINAL command, 8–20
/SCRATCH switch
  SAVRES, E–6
/SEIZE qualifier
  ALLOCATE command, 11–13
/SELECT qualifier
  BACKUP command, 16–9
  RESTORE command, 16–17
/SELECT_ERASE qualifier
  SET TERMINAL command, 8–20
Send/receive quota, 5–4
  message, 5–4
  RIB (Receiver ID Block), 5–4
Send/Receive quotas, 5–1
SEND privilege
  BROADCAST command, 6–5
  restricted receiver, 6–5
Sequential data caching, 11–8, 11–9
  example, 11–9
Servers
  assigned to queues, 9–4
SET ACCOUNT command, 7–13
SET CACHE command
  /ALL qualifier, 11–18
  /CLUSTERSIZE qualifier, 11–19
  /DATA_MAXIMUM qualifier, 11–20
  /DIRECTORY_MAXIMUM qualifier, 11–20
  /DISABLE qualifier, 11–20
  /FILE qualifier, 11–18
  /KEEP qualifier, 11–20
  /MAXIMUM qualifier, 11–19
  /NOFILE qualifier, 11–18
  qualifiers, 11–18
  TUNE privilege, 11–18
SET DATE command, 14–3, 14–24
SET DEVICE command
  /DISABLE qualifier, 11–1
  /ENABLE qualifier, 11–1
  /NORESTRICT qualifier, 11–1
  /RESTRICT qualifier, 11–1
SET FILE/RUNTIME_SYSTEM=name command, 12–6
SET FILE command, 12–35
SET JOB/HOLD command, 13–5
  SET JOB/RELEASE, 13–5
SET JOB/PRIORITY command, 13–6
SET JOB/RUN_BURST command, 13–6
SET JOB/SIZE command, 13–7
SET JOB command, 13–17
SET NODE/LAT command, 8–12
SETPAS privilege
  SET PASSWORD command, 6–5
SET PASSWORD command, 7–16
  SETPAS privilege, 6–5
SET PRIVILEGE command, 6–1
SET QUEUE command, 9–32
SET SERVER command, 9–35
SET SERVICE/LAT command, 8–13

SET SYSTEM/ANSWERBACK command, 14–21
SET SYSTEM/DYNAMIC_REGION_LIMIT command, 5–4
SET SYSTEM/EMT_LOGGING command, 14–4, 14–26
SET SYSTEM/HOLD command, 14–2
SET SYSTEM/LOGINS command, 12–15, 13–1, 13–2
SET SYSTEM/MONITOR_NAME command, 2–6
SET SYSTEM/NOLOGINS command, 13–1
SET SYSTEM/RELEASE command, 14–2
SET SYSTEM command, 8–14, 13–19, 14–25
SET TERMINAL/ANSWERBACK/PERMANENT command, 14–21
SET TERMINAL command, 8–15
  /AUTOBAUD qualifier, 8–1
  /BROADCAST qualifier, 8–6
  dial-up lines, 8–1
  HWCFG privilege, 8–3
  /INQUIRE, 8–8
  /NOBROADCAST qualifier, 8–6
  /PERMANENT qualifier, 8–5
  START.COM, 8–1, 8–5
SET TIME command, 14–3, 14–29
Setting job priority, 13–5
Setting job size, 13–5
/SETUP qualifier
  SET TERMINAL command, 8–20
/SHAREABLE qualifier
  INITIALIZE/SERVER command, 9–30
  INSTALL/LIBRARY command, 12–24
  INSTALL/RUNTIME_SYSTEM command, 12–27
  SET SERVER command, 9–36
SHOW ACCOUNT command, 7–18
  sample output, 7–18
SHOW BUFFERS command, buffer status report, 12–15
SHOW CACHE command, 11–26
SHOW COMMAND/SYSTEM command, 12–12, 12–37
SHOW COUNTERS/LAT command, 8–23
SHOW DEVICE command, 11–28
SHOW DEVICES/ALLOCATED
  use for shutdown, 4–1
SHOW DISKS
  use for shutdown, 4–1
SHOW DISKS command
  free space, 14–8
SHOW FILE/SYSTEM command, 12–38
SHOW JOB/PRIVILEGE command, 13–20
SHOW JOBS
  use for shutdown, 4–1
SHOW JOBS/DETACHED command, 14–8
SHOW JOBS command
  job status, 14–8
SHOW LIBRARY command, 12–39
SHOW LOGICAL/SYSTEM command, 12–10, 12–41
SHOW LOGICALS command, 12–40
SHOW NODE/LAT command, 8–25
SHOW RUNTIME_SYSTEM command, 12–42
SHOW SERVER command, 9–37
SHOW SERVICE/LAT command, 8–26
SHOW SESSIONS command, 8–27
SHOW SYSTEM/EMT_LOGGING command, 14–5, 14–30
SHOW SYSTEM command, 14–8, 14–30
  monitor name, 2–6
  swap size, 2–6
SHOW TERMINAL command, 8–6, 8–29
  sample output, 8–29
SHOW TERMINAL_SERVERS/LAT command, 8–31
SHOW USER command
  job control, 13–4
SHOW USERS command
  job status, 14–8
Shutdown
  disabling DECnet/E during, 4–4
  ERRCPY, 4–9
  EVTLOG program, 4–8
  final, under SHUTUP, 4–10
  formula for waiting period, 4–5
  manual procedure, A–44
  normal (example), 4–10
  OPSER, 4–8, A–5
  performing system, 4–1
  RSTS/E system, 4–1
  specify waiting period, 4–3
SHUTUP
  attached jobs, 4–6
  automatic restart, 4–4
  communicate with OPSER, 4–2
  detached jobs, 4–6
  disable DECnet/E, 4–4
  disabling logins, 4–4
  disk dismount phase, 4–10
  EMT logging shutdown phase, 4–9
  ERRCPY shutdown phase, 4–9
  error message to ERRCPY, 10–3
  example
    DECnet/E shutdown, 4–12
  final job killing phase, 4–9
  final shutdown phase, 4–10
  formula for waiting period, 4–5
  initial job killing phase, 4–6
  limit logins, 4–4
  limit network activity, 4–4
  operational phases, 4–3
  OPSER shutdown phase, 4–8
  perform shutdown with, 4–1
  phases of, 4–2
  removing swapping files, 12–15
  resident library removal phase, 4–9
  run from console terminal, 4–2
  running QUEMAN after, A–20
  run-time system removal phase, 4–9
  run with OPSER, 4–3
  run without OPSER, 4–3
  set up dialogue phase, 4–2
  shutdown of DECnet/E, 4–4
  specify shutdown wait period, 4–3
  swap file removal phase, 4–10
  system recovery, 16–2
  terminating OPSER, A–5, A–44
  treatment of PBS, 4–6
  unexpected errors, 4–10
  use of, 2–3
  warning message, 4–5
SHUTUP example
  large RSTS/E system, 4–10
  small RSTS/E system, 4–12
SHUTUP privilege
  system control, 6–5
SIL (Save Image Library)
  format, 12–7
  MAKSIL program, 12–7

Simulation of hardware errors
  device testing package, F–1
/SIXEL qualifier
  SET TERMINAL command, 8–21
/SIZE qualifier
  CREATE/ACCOUNT command, 7–9
  INSTALL/ERROR_FILE command, 12–22
  INSTALL/OVERLAY_FILE command, 12–25
  INSTALL/SWAP_FILE command, 12–29
  SET JOB command, 13–18
Small buffers
  DISPLY report, 14–18
  general (SYSTAT), 14–8
  SHOW BUFFERS command, 14–8
Software errors
  logging, 10–1
/SPECIAL_CHARACTER qualifier
  SET PRINTER command, 11–24
/SPEED qualifier
  SET TERMINAL command, 8–21
SPOOL, A–1
  ABORT command, A–26, A–35
  /ALIGN switch, A–37
  ASSIGN option, A–24
  changing NORMAL form, A–24
  compiled version of, A–21
  default form, A–24
  default keyboard values, A–28
  default line printer values, A–27
  /DFLENGTH switch, A–25
  error handling, A–34
  error messages, A–36
  error text in user output, A–36t
  file header, A–35
  FORM command, A–36
  FORM option, A–24
  forms control characters, A–25
  interrupt commands, A–30, A–30t
  job header, A–35
  line printer errors, A–34
  logical device name, A–22
  modules, A–21
  NAME option, A–24
  NORMAL option, A–24
  /PAGE_EJECT switch, A–25
  PHYSICAL option, A–24
  QUE command, A–35
  raise priority, A–32
  raise run burst, A–32
  REQUE command, A–34
  RESTART command, A–34
  restart queued request, A–34
  run, A–21
  startup
    error processing, A–28
    examples, A–31
    options, A–23t
  syntax error messages, A–28, A–29t
  terminate
    queued request, A–35
    without shutting down, A–44
Spooling
  determining number of jobs, A–18
  initial conditions for, queues, A–14
  keyboard, A–28
  line printer, A–27
Spooling Package Library, A–21
Spooling programs
  commands, A–12
  controlling, A–12
  line printer, A–21
  messages, A–12
  monitoring, A–12
  SPOOL, A–21
  SPOOL and BATCH, A–1
  table of in QUEMAN, A–4
START.COM
  automatic restart, 2–4
  memory image, 2–5
  sample, 3–2
  SET TERMINAL command, 8–1, 8–5
START.COM file
  ANALYS, 10–23
  commands auxiliary run-time system, 12–4
  contents, 3–1
  logical names, 12–9
  replace, 3–1
START/LAT command, 8–32
START/QUEUE/MANAGER command, 9–40
START/QUEUE command, 9–39
START/SERVER command, 9–42
Startup
  BATCH options, A–39t
  BATCH procedures, A–41
  CCL command definitions, 3–8
  controlling system, 3–1
  SPOOL options, A–23t
  terminal characteristics, 3–8
Startup control file
  order of operations, 3–2
/STATIC_RATING qualifier
  CREATE/SERVICE/LAT command, 8–10
  SET SERVICE/LAT command, 8–13
Statistics
  abbreviations in DISPLY report, 14–14
  busy device, 14–16
  job status, 14–14
  SAVRES run total, E–23t
STATUS command
  QUEMAN, A–18
STOP/LAT command, 8–33
STOP/QUEUE/MANAGER command, 9–45
STOP/QUEUE command, 9–44
STOP/SERVER command, 9–46
Summary report
  ERRDIS, 10–6
  run totals, E–23t
SWAP0.SYS file
  allocate and position, 12–13
SWAP1.SYS file
  allocate and position, 12–13
SWAP3.SYS file
  allocate and position, 12–13
Swap file, 1–12
  add, 12–13
  affects logins, 12–15
  online creation, 12–14
  removing, 12–13, 12–15
    under SHUTUP, 4–10, 12–15
  SHOW FILE/SYSTEM command, 12–15
Swapping, 1–12
Swapping mode
  TUNE privilege, 6–6
Swap space, 13–2

Swap space (cont'd.)
restricting logins, 12–15
/SWAP_FILE qualifier
SHOW FILE/SYSTEM command, 12–38
/SWAP_MAXIMUM qualifier
SET SYSTEM command, 14–28
SWCFG privilege
system manager function, 6–5
SWCTL privilege
EMT logging, 14–5, 14–26
Logins, 6–5
Network management, 6–5
SWITCH program
change keyboard monitor, 1–1
SY0:
system disk, 1–2
SYS functions
programming, 2–5
SYSIO privilege
file access, 6–5
write access, 6–7
SYSMOD privilege
system modification, 6–6
SYSTAT
free disk space, 14–8
guidelines, 14–8
use, 14–7
System
automatic restart RSTS/E, 2–2
commands, 1–11
EMT logging to record activities, 14–4
enable data caching, 11–18
enable directory caching, 11–18
halt, after bootstrapping RSTS/E, 2–2
halting the RSTS/E, 2–3
initialization options, 1–2
initialized in restart mode, 2–5
monitoring status, 14–8
operational concepts, 1–11
shutdown, 4–1
startup, 3–1
suspending operations, 14–2
System clock
DATES privilege, 6–3
System commands
INSTAL privilege, 6–4
System control
HWCTL privilege, 6–4
JOBCTL privilege, 6–4
SHUTUP privilege, 6–5
System crash, 2–2
ANALYS program, 3–9
causes, 2–4
handling, 2–5
randomly occurring, 2–5
System disk, 1–2
access, 1–3
contents, 1–3
recovery, 16–2
System error log, F–1
System Error Package
programs, 10–1
System errors
compilation, 10–1
extraction, 10–1
formatting, 10–1
retention, 10–1
System files
INSTAL privilege, 6–4
System Files
managing with DCL commands, 12–1t
System logical names
INSTAL privilege, 6–4
System manager function
HWCFG privilege, 6–3
SWCFG privilege, 6–5
System modification
SYSMOD privilege, 6–6
System programs, 1–2
/SYSTEM qualifier
SET PASSWORD command, 7–17
System restart
START.COM, 2–5
System security
EMT logging, 14–7
System startup
command file, 3–2
conditions at, 3–1
controlling, 3–1
crash recovery, 3–9

T

/TAB qualifier
SET PRINTER command, 11–24
SET TERMINAL command, 8–21
Tape handling, DCL commands, 15–1t
/TEMPLATE qualifier
CREATE/ACCOUNT command, 7–9
TERDFL.SYS
terminal characteristics, 8–4
TERMAC.SYS
terminal characteristics, 8–4
Terminal
at system command level, 1–11
/BROADCAST command qualifier, 8–6
Ctrl/C typed at, 1–11
/DFLENGTH switch effect, A–26
distinction from line printer, A–27
FORCE command, 13–3
hardware top of form, A–28
line speed characteristics file, 8–2
/NOBROADCAST command qualifier, 8–6
OSC, A–13
setting characteristics, 8–1
top of form capability, A–26
Terminal characteristics
HWCFG privilege, 8–3
Startup, 3–8
TERDFL.SYS, 8–4
TERMAC.SYS, 8–4
Terminal device designator, 8–2
Terminal line
dial-up lines, 8–1
local, 8–1
Terminal servers, 3–9, 8–6
/TERMINAL_SERVERS qualifier
SHOW COUNTERS/LAT command, 8–24
TERSPD.SYS file
account location, 8–2
/AUTOBAUD qualifier, 8–2
why create, 8–2
Timesharing, start, 2–3

/TIME_FORMAT qualifier
SET SYSTEM command, 14–28
/TIME_LIMIT qualifier
INITIALIZE/QUEUE command, 9–27
SET QUEUE command, 9–34
TMPPRV privilege
set privilege bit, 6–6
/TOP_OF_FILE qualifier
START/QUEUE/MANAGER command, 9–43
/TTSYNC qualifier
SET TERMINAL command, 8–21
TUNE privilege
alter base priority, 6–6
SET CACHE command, 11–18
swapping mode, 6–6
swap size, 2–6
/TYPE qualifier
INSTALL/RUNTIME_SYSTEM command, 12–27
SET TERMINAL command, 8–21

U

UFD (User File Directory)
and MFD in Level 0, 1–5f
mark file for caching, 11–8
reorder, 11–5
with MFD and GFDs in Level 1, 1–6f
UNLOAD/INDEX command, 14–31
UNLOAD/LIBRARY command, 12–8, 12–43
UNLOAD/OVERLAY command, 14–32
UNLOAD/RUNTIME_SYSTEM command, 12–6, 12–44
/UNLOAD qualifier
DISMOUNT command, 15–5
INSTALL/LIBRARY command, 12–24
INSTALL/RUNTIME_SYSTEM command, 12–27
/UPPERCASE qualifier
SET PRINTER command, 11–25
SET TERMINAL command, 8–21
/UP_ARROW qualifier
SET TERMINAL command, 8–21
USER1-8 privileges
User-defined privileges, 6–8
User account, 7–1
User-defined privileges
USER1-8 privileges, 6–8
User interface
Print/Batch Services (PBS), 9–2
User job
area, 1–11
/USER qualifier
CREATE/ACCOUNT command, 7–9
DELETE/ACCOUNT command, 7–11
SET ACCOUNT command, 7–15
SHOW ACCOUNT command, 7–20
/USER_DEFINED_KEYS qualifier
SET TERMINAL command, 8–22
UTILTY program
KILL, A–45

V

/VERIFY qualifier
BACKUP command, 16–9
RESTORE command, 16–17
Virtual disks, 11–12
Volumes
dismounting SAVRES, E–20
mounting SAVRES, E–20

W

WACNT privilege
account management, 6–7
/WIDTH qualifier
SET PRINTER command, 11–25
SET TERMINAL command, 8–22
Window turning
reducing, 11–11
Work file
OPSER stores, A–2
WREAD privilege
execute access, 6–7
read access, 6–7
Write access
GWRITE privilege, 6–3
SYSIO privilege, 6–7
WWRITE privilege, 6–7
/WRITE qualifier
INITIALIZE command, 15–11
WRTNFS privilege
non-file-structured I/O, 6–7
WWRITE privilege
create access, 6–7
rename access, 6–7
write access, 6–7

X

XBUF
caching allocation, 11–7, 11–9, 11–10
resident library allocation, 11–10

Z

Zeroing the ERRDIS log, 10–20
/ZERO qualifier
SHOW COUNTERS/LAT command, 8–24

Reader's Comments

RSTS/E
System Manager's Guide
AA-2762G-TC

Please use this postage-paid form to comment on this manual. If you require a written reply to a software problem and are eligible to receive one under Software Performance Report (SPR) service, submit your comments on an SPR form.

Thank you for your assistance.

I rate this manual's:	Excellent	Good	Fair	Poor
Accuracy (software works as manual says)	□	□	□	□
Completeness (enough information)	□	□	□	□
Clarity (easy to understand)	□	□	□	□
Organization (structure of subject matter)	□	□	□	□
Figures (useful)	□	□	□	□
Examples (useful)	□	□	□	□
Index (ability to find topic)	□	□	□	□
Page layout (easy to find information)	□	□	□	□

I would like to see more/less ____

What I like best about this manual is ____

What I like least about this manual is ____

I found the following errors in this manual:

Page Description

Additional comments or suggestions to improve this manual:

I am using **Version** ____ of the software this manual describes.

Name/Title ____ Dept. ____
Company ____ Date ____
Mailing Address ____
____ Phone ____

› Not Tear - Fold Here and Tape

digital™

No Postage
Necessary
if Mailed
in the
United States

BUSINESS REPLY MAIL
FIRST CLASS PERMIT NO. 33 MAYNARD MASS.

POSTAGE WILL BE PAID BY ADDRESSEE

Digital Equipment Corporation
Corporate User Publications
Continental Boulevard
MK01-2/E12
Merrimack, New Hampshire 03054

o Not Tear - Fold Here