

Installing and Managing Printers on the DG/UX™ System

093-701132-01

Installing and Managing Printers on the DG/UX™ System

093-701132-01

For the latest enhancements, cautions, documentation changes, and other information on this product, please see the Release Notice (085-series) and/or Update Notice (078-series) supplied with the software.

Copyright ©Data General Corporation, 1994
All Rights Reserved
Unpublished – all rights reserved under the copyright laws of the United States.
Printed in the United States of America
Rev. 01, July 1994
Licensed Material – Property of Data General Corporation
Ordering No. 093-701132

Notice

DATA GENERAL CORPORATION (DGC) HAS PREPARED THIS DOCUMENT FOR USE BY DGC PERSONNEL, LICENSEES, AND CUSTOMERS. THE INFORMATION CONTAINED HEREIN IS THE PROPERTY OF DGC; AND THE CONTENTS OF THIS MANUAL SHALL NOT BE REPRODUCED IN WHOLE OR IN PART NOR USED OTHER THAN AS ALLOWED IN THE DGC LICENSE AGREEMENT.

DGC reserves the right to make changes in specifications and other information contained in this document without prior notice, and the reader should in all cases consult DGC to determine whether any such changes have been made.

THE TERMS AND CONDITIONS GOVERNING THE SALE OF DGC HARDWARE PRODUCTS AND THE LICENSING OF DGC SOFTWARE CONSIST SOLELY OF THOSE SET FORTH IN THE WRITTEN CONTRACTS BETWEEN DGC AND ITS CUSTOMERS. NO REPRESENTATION OR OTHER AFFIRMATION OF FACT CONTAINED IN THIS DOCUMENT INCLUDING BUT NOT LIMITED TO STATEMENTS REGARDING CAPACITY, RESPONSE-TIME PERFORMANCE, SUITABILITY FOR USE OR PERFORMANCE OF PRODUCTS DESCRIBED HEREIN SHALL BE DEEMED TO BE A WARRANTY BY DGC FOR ANY PURPOSE, OR GIVE RISE TO ANY LIABILITY OF DGC WHATSOEVER.

This software is made available solely pursuant to the terms of a DGC license agreement, which governs its use.

Restricted Rights: Use, duplication, or disclosure by the U. S. Government is subject to restrictions as set forth in subparagraph (c)(1)(ii) of the Rights in Technical Data and Computer Software clause at Defense Federal Acquisition Regulation (DFARS) 252.227-7013 and in subparagraphs (a) through (d) of the Commercial Computer Software Restricted Rights clause at Federal Acquisition Regulations (FAR) 52.227-19, whichever may apply.

Data General Corporation
4400 Computer Drive
Westboro, MA 01580

AV Object Office, AV Office, AViON, CEO, CLARiON, DASHER, DATAPREP, DESKTOP GENERATION, ECLIPSE, ECLIPSE MV/4000, ECLIPSE MV/6000, ECLIPSE MV/8000, GENAP, INFOS, microNOVA, NOVA, OpenMAC, PRESENT, PROXI, SWAT, TRENDVIEW, and WALKABOUT are U.S. registered trademarks of Data General Corporation; and AOSMAGIC, AOS/VSMAGIC, AROSE/PC, ArrayPlus, AV Image, AV Imager Toolkit, AV SysScope, BaseLink, BusiGEN, BusiPEN, BusiTEXT, CEO Connection, CEO Connection/LAN, CEO Drawing Board, CEO DXA, CEO Light, CEO MAIL, CEO Object Office, CEO PXA, CEO Wordview, CEOwrite, COBOL/SMART, COMPUCALC, CSMAGIC, DATA GENERAL/One, DESKTOP/UX, DG/500, DG/AROSE, DGConnect, DG/DBUS, DG/Fontstyles, DG/GATE, DG/GEO, DG/HEO, DG/L, DG/LIBRARY, DG/UX, DG/Vision, DG/XAP, ECLIPSE MV/1000, ECLIPSE MV/1400, ECLIPSE MV/2000, ECLIPSE MV/2500, ECLIPSE MV/3200, ECLIPSE MV/3500, ECLIPSE MV/3600, ECLIPSE MV/5000, ECLIPSE MV/5500, ECLIPSE MV/5600, ECLIPSE MV/7800, ECLIPSE MV/9300, ECLIPSE MV/9500, ECLIPSE MV/9600, ECLIPSE MV/10000, ECLIPSE MV/15000, ECLIPSE MV/18000, ECLIPSE MV/20000, ECLIPSE MV/25000, ECLIPSE MV/30000, ECLIPSE MV/35000, ECLIPSE MV/40000, ECLIPSE MV/60000, FORMA-TEXT, GATEKEEPER, GDC/1000, GDC/2400, Intellibook, microECLIPSE, microMV, MV/UX, OpStar, PC Liaison, RASS, REV-UP, SLATE, SPARE MAIL, SUPPORT MANAGER, TEO, TEO/3D, TEO/Electronics, TURBO/4, UNITE, and XODIAC are trademarks of Data General Corporation. AV/Alert is a service mark of Data General Corporation.

UNIX is a U.S. registered trademark of Unix System Laboratories, Inc. BSD is a trademark of the Regents of the University of California, Berkeley. NFS is a U.S. registered trademark and ONC is a trademark of Sun Microsystems, Inc. The Network Information Service (NIS) was formerly known as Sun Yellow Pages. The functionality of the two remains the same; only the name has changed. The name Yellow Pages is a registered trademark in the United Kingdom of British Telecommunications plc and may not be used without permission. SUNOS is a trademark of Sun Microsystems, Inc. Novell and NetWare are U.S. registered trademarks of Novell, Inc. DEC is a trademark of Digital Equipment Corporation. PostScript is a U.S. registered trademark of Adobe Systems, Inc. MS-DOS is a U.S. registered trademark of Microsoft Corporation. MILAN and FastPort are trademarks of MILAN Technology Corporation. HP and LaserJet are trademarks of Hewlett-Packard. LaserJet IIISi is a trademark of Hewlett-Packard. Centronics is a registered trademark of Centronics Corporation. Epson FX is a trademark of Epson America, Inc. ProPrinter, ProPrinter II, and ProPrinter XL are U.S. registered trademarks of International Business Machines Corporation. OKIDATA and MICROLINE are U.S. registered trademarks of Oki Electric Industry Co., Ltd. Printronix is a U.S. registered trademark of Printronix Incorporated. Dataproducts is a U.S. registered trademark of Dataproducts Corporation. MT660/690 is a U. S. registered trademark of Mannesmann Tally Corporation.

Installing and Managing Printers on the DG/UX™ System

093-701132-01

Revision History:

Original Release –January 1994

First Revision –July 1994

Effective with:

DG/UX 5.4, Release 3.00

DG/UX 5.4, Release 3.10

A vertical bar in the margin of a page indicates substantive technical change from the previous revision. (The exceptions are Chapters 7, 8, and 9, which are new to this manual.)

About this manual

This manual describes how to install and manage printers on the DG/UX™ system. It contains information on cabling, stty options, and emulation modes. This manual explains how to connect printers to hosts, define printers in the DG/UX system, manage the LP subsystem, and troubleshoot problems.

This manual is intended for users who install, set up, and manage printers on DG/UX systems and networks. The manual assumes these users are computer knowledgeable with some experience in system administration or operation.

How this manual is organized

This manual contains nine chapters and three appendixes. The following list gives an overview of what you will find in these chapters and appendixes:

- Chapter 1 Introduces the user to general concepts on the printing process. It describes printer characteristics, methods of connecting printers to hosts, methods for setting up the line printer spooler, and document types.
- Chapter 2 Explains issues on connecting printers to hosts and network devices. It describes asynchronous controllers, termservers, MiLAN™ print servers, and interface cables.
- Chapter 3 Explains how to configure local printers. It describes printers connected to `/dev/lp`, Duart ports, and Syac controllers through `/dev/tty`.
- Chapter 4 Explains how to configure printers connected to a remote host running the DG/UX system 5.4 or greater and the lpNet daemon. It also describes printers connected to a remote host that supports the Line Printer Daemon Protocol (LPD).
- Chapter 5 Explains how to configure printers connected to DG/UX systems prior to 5.4 and AOS/VS II systems.
- Chapter 6 Explains how to configure printers connected to the network by termservers and MiLAN print servers.
- Chapter 7 Explains the `sysadm` Printer menu operations and LP services concepts in detail.

- Chapter 8 Explains the shell-level commands that **sysadm** uses to provide LP services.
- Chapter 9 Explains how to maintain print services with the LP print service utilities and discusses how to handle PostScript® printers.
- Appendix A Contains troubleshooting tips and explains two methods of troubleshooting: the first uses a defined set of steps; the second examines the flow of data—the Data Flow Isolation technique (DFI). This appendix also includes helpful information about terminfo, streams, MiLAN FastPort™ devices, term servers, print filters, remote printing and printing with BSD.
- Appendix B Explains how to determine tty lines for asynchronous ports. It describes how to determine the order of terminal line controllers listed in the system file, how to use the AViON® System Diagnostics, and how to determine the tty line number for each device on a terminal line controller.
- Appendix C Contains worksheets for recording information about local and remote printers.

Related Data General manuals

Within this manual, we refer to the following manuals:

Installing the DG/UX™ System (093–701087). This manual describes how to install the DG/UX operating system on AViON hardware.

Managing the DG/UX™ System (093–701088). This manual discusses the concepts and tasks related to DG/UX system management. It approaches system administration through the **sysadm** facility.

Managing ONC™/NFS® and Its Facilities on the DG/UX™ System (093–701049). This manual explains how to manage and use the DG/UX ONC™/NFS® product. It contains information on the Network File System (NFS), the Network Information Service (NIS), Remote Procedure Calls (RPC), and External Data Representation (XDR).

HPS Downloadable Cluster Controller Installation Guide (014–001814). This manual is the primary source for installing, removing, and connecting cluster controllers in conjunction with VME distributed adapters (VDA/128 or VDA/255) in AViON systems. Use this manual for both Model 20 and Model 30 cluster controllers.

Managing TCP/IP on the DG/UX™ System (093–701051). This manual explains how to prepare for setup of Data General's TCP/IP (DG/UX) package on AViON computer systems. It tells how to tailor the software at your site, and use **sysadm** to manage the package, and troubleshoot system problems.

Setting Up and Installing VMEbus Options in AViON® Systems (014–001867). This manual describes how to jumper VMEbus controllers to operate in an AViON environment. Explains how to install and remove the 6u controller boards in the VMEbus card cages. Supplies instructions for connecting external devices to the controller boards.

Other related documents

This section lists other related documents, which are not available from Data General Corporation.

- *FastPort™ 3000 and 3100 Network Print Server User's Guide* Version 2.5. MiLAN™ Technology Corporation, October, 1992.
- *NetWare NFS Supervisor's Guide* Second Edition. Novell, Inc., December 1991.

Reader, please note:

Data General manuals use certain symbols and styles of type to indicate different meanings. The Data General symbol and typeface conventions used in this manual are defined in the following list. You should familiarize yourself with these conventions before reading the manual.

This manual also presumes the following meanings for the terms "command line," "format line," and "syntax line." A command line is an example of a command string that you should type verbatim; it is preceded by a system prompt and is followed by a delimiter such as the curved arrow symbol for the New Line key. A format line shows how to structure a command; it shows the variables that must be supplied and the available options. A syntax line is a fragment of program code that shows how to use a particular routine; some syntax lines contain variables.

Throughout this manual we use the following format conventions:

Conventions	Meaning
boldface	In command lines and format lines: Indicates text (including punctuation) that you type verbatim from your keyboard.

All DG/UX commands, pathnames, and names of files, directories, and manual pages also use this typeface.

Typewriter

Represents a system response on your screen. Syntax lines also use this font.

italic

In format lines: Represents variables for which you supply values; for example, the names of your directories and files, your username and password, and possible arguments to commands.

In text: Indicates a term that is defined in the manual's glossary.

[]

In format lines: These brackets surround an optional argument. Don't type the brackets; they only set off what is optional. The brackets are in regular type and should not be confused with the boldface brackets shown below.

[]

In format lines: Indicates literal brackets that you should type. These brackets are in boldface type and should not be confused with the regular type brackets shown above.

...

In format lines and syntax lines: Means you can repeat the preceding argument as many times as desired.

\$ and %

In command lines and other examples: Represent the system command prompt symbols used for the Bourne and C shells, respectively. Note that your system might use different symbols for the command prompts.

)

In command lines and other examples: Represents the system command prompt symbol used for the AOS/VS II CLI.

↵

In command lines and other examples: Represents the New Line key, which is the name of the key used to generate a new line. (Note that on some keyboards this key might be called Enter or Return instead of New Line.) Throughout this manual, a space precedes the New Line symbol; this space is used only to improve readability—you can ignore it.

- < >** In command lines and other examples: Angle brackets distinguish a command sequence or a keystroke (such as **<Ctrl-D>**, **<Esc>**, and **<3dw>**) from surrounding text. Note that these angle brackets are in regular type and that you do not type them; there are, however, boldface versions of these symbols (described below) that you do type.
- <, >, >>** In text, command lines, and other examples: These boldface symbols are redirection operators, used for redirecting input and output. When they appear in boldface type, they are literal characters that you should type.
- In command lines and other examples: Represents the cursor, which indicates your current typing position on the screen.

Contacting Data General

Data General wants to assist you in any way it can to help you use its products. Please feel free to contact the company as outlined below.

Manuals

If you require additional manuals, please use the enclosed TIPS order form (United States only) or contact your local Data General sales representative.

Telephone assistance

If you are unable to solve a problem using any manual you received with your system, free telephone assistance is available with your hardware warranty and with most Data General software service options. If you are within the United States or Canada, contact the Data General Customer Support Center (CSC) by calling 1-800-DG-HELPS. Lines are open from 8:00 a.m. to 5:00 p.m., your time, Monday through Friday. The center will put you in touch with a member of Data General's telephone assistance staff who can answer your questions.

For telephone assistance outside the United States or Canada, ask your Data General sales representative for the appropriate telephone number.

Joining our users group

Please consider joining the largest independent organization of Data General users, the North American Data General Users

Group (NADGUG). In addition to making valuable contacts, members receive *FOCUS* monthly magazine, a conference discount, access to the Software Library and Electronic Bulletin Board, an annual Member Directory, Regional and Special Interest Groups, and much more. For more information about membership in the North American Data General Users Group, call 1-800-253-3902 or 1-508-443-3330.

End of Preface

Contents

Chapter 1 – Setting up printers and queues

Identifying physical printer characteristics	1-1
Printer emulations	1-1
Choosing the printer port	1-2
Setting up printers using a parallel port	1-2
Setting up printers using a serial port	1-3
Setting up queues in the DG/UX system	1-4
Software configurable parameters	1-4
Hardware configurable parameters	1-8
Adding printers with multiple emulation modes	1-9
Understanding document types	1-10

Chapter 2 – Connecting printers

Interconnectivity concepts	2-1
Connecting to hosts	2-3
Connecting to a Duart port	2-3
Connecting to VDA/VAC controllers	2-4
Connecting to network devices	2-5
Connecting to Term servers	2-5
Connecting to VTC boards	2-10
Connecting to MiLAN FastPort print servers	2-13
Interface cables	2-14
Interface cables listed by device-type and purpose	2-16
Cabling diagrams for serial printers	2-17
Parallel printer cables	2-21

Chapter 3 – Configuring local printers

Adding printers connected to /dev/lp	3-2
Adding a line printer connected to /dev/lp	3-4
Adding a PostScript printer connected to /dev/lp	3-5
Adding printers connected to a Duart	3-7
Adding an Epsonfx printer connected to a Duart	3-8
Adding printers connected to a Syac	3-10
Adding a PostScript printer connected to a Syac	3-12
Adding local printers on a Trusted DG/UX system	3-13
Adding a PostScript printer connected to /dev/lp	3-14

Chapter 4 – Configuring remote printers connected via lpNet and LPD

Adding remote printers connected to DG/UX systems and LPD systems	4-2
---	-----

Allowing host access to remote printer resources	4-2
Adding a remote PostScript printer using lpNet	4-4
Adding printers connected to Novell NetWare systems	4-6
Configuring Novell systems for remote printing	4-7
Configuring the DG/UX system for printing to NetWare systems	4-8
Adding a remote PostScript printer connected to a NetWare system	4-12
Adding printers connected to SunOS systems	4-13
Configuring SunOS systems for remote printing	4-13
Configuring the DG/UX system for printing to SunOS systems	4-14
Adding a remote line printer connected to a SunOS system	4-16
Adding remote printers connected to DG/UX systems and LPD systems	
on a Trusted DG/UX system	4-17
Adding a remote PostScript printer using lpNet	4-18

Chapter 5 – Configuring remote printers using the remshp program

Adding printers attached to DG/UX systems prior to Release 5.4	5-1
Enabling remote printer access on DG/UX systems prior to 5.4	5-1
Adding a remote printer attached to a DG/UX system prior to 5.4	
to your local system	5-3
Adding a remote HP LaserJet printer	5-4
Default remote remsh printer attributes	5-6
Adding remote printers attached to AOS/VS II systems	5-7
Setting up AOS/VS II systems for remote printing	5-7
Adding printers from a remote AOS/VS II system	5-9
Adding a remote PostScript printer	5-10
Default attributes for remote remsh printers on AOS/VS II systems	5-13
Adding printers attached to DG/UX systems prior to Release 5.4	
on a Trusted DG/UX system	5-14
Adding a remote HP LaserJet printer on a Trusted system	5-15

Chapter 6 – Configuring remote printers connected to the network

Printers connected to termserverns	6-2
Adding printers to termserverns	6-2
Adding a plotter connected to a termserver	6-4
Adding a PostScript printer connected to an annex termserver	6-5
Adding HP LaserJet 4 printers with HP JetDirect	6-7
Adding an HP LaserJet 4 printer	6-9
Printers connected to FastPort print servers	6-11
Adding printers connected to FastPort print servers	6-12
Adding a PostScript printer connected to a FastPort print server	6-13
Remote networked printers on a Trusted DG/UX system	6-15
Adding a PostScript printer connected to a FastPort print server	6-16

Chapter 7 – sysadm management menus

LP management procedures	7-1
Managing printer devices	7-2
Managing classes	7-23
Managing filters	7-25
Managing Forms	7-29
Setting User and Request Priorities	7-34
Managing Requests	7-36
Managing the Scheduler	7-37
Managing Remote Systems	7-37
Displaying LP Service Status	7-39

Chapter 8 – Configuring printers from the shell

Overview	8-2
Components of a print service	8-2
Functions performed by the print service software	8-2
Suggestions for LP print service administration	8-3
Configuring printer sites	8-3
Configuring printers	8-4
Printer name	8-5
Connection method	8-6
Interface program	8-10
Printer types	8-11
Content types	8-11
Printer port characteristics	8-14
Character sets or print wheels	8-16
Alerting to mount a print wheel	8-18
Forms allowed	8-20
Printer fault alerting	8-21
Printer fault recovery	8-23
User access restrictions	8-24
Banner pages	8-25
Printer description	8-26
Default printing attributes	8-26
Printer class membership	8-27
System default destination	8-28
Mounting a form or print wheel	8-29
Removing a printer or class	8-30
Putting it all together	8-31
Examining a printer configuration	8-32
Making printers available	8-33
Accepting print requests for a new printer	8-33
Enabling and disabling a printer	8-33

Chapter 9 – Administering the LP subsystem

Providing forms	9-2
Defining a form	9-2
Removing a Form	9-5
Restricting user access	9-6
Alerting to mount a form	9-7
Mounting a form	9-9
Examining a form	9-9
Providing filters	9-9
What is a filter?	9-10
Filters for your system	9-14
Removing a filter	9-23
Examining a filter	9-23
Restoring factory defaults	9-23
A word of caution	9-23
Managing the printing load	9-24
Rejecting requests for a printer or class	9-24
Accepting requests for a printer or class	9-25
Moving requests to another printer	9-25
Examples	9-26
Managing queue priorities	9-27
Setting priority limits	9-28
Setting a default priority	9-28
Examining the priority limits and defaults	9-29
Moving a request around in the queue	9-29
Starting and stopping the LP print service	9-30
Manually stopping the print service	9-31
Manually starting the print service	9-31
Managing the LP print service logs	9-32
The lpNet log	9-32
The lpsched log	9-32
The requests log	9-33
Managing PostScript printers	9-35
How to use a PostScript printer	9-36
Support of non-PostScript print requests	9-36
Additional PostScript capabilities provided by filters	9-37
The administrator's duties	9-38
Customizing the print service	9-46
Adjusting the printer port characteristics	9-48
Adjusting the terminfo database	9-49
How to write an interface program	9-52
How to write a filter	9-57
Translating character sets	9-58
Managing the LP subsystem on a trusted DG/UX system	9-62

Labeling output	9-62
Access control	9-66
Auditing the LP subsystem	9-67

Appendix A – Troubleshooting

Quick troubleshooting tips	A-1
No Output (Nothing is Printed)	A-1
Illegible Output	A-2
Legible Printing, but Wrong Spacing	A-3
Wrong Characters Printed	A-4
Dial Out Failures	A-5
Idle Printers	A-5
Networking Problems	A-6
Troubleshooting printer problems in lpsched	A-7
Verifying the physical connection and printer setup	A-8
Verifying the printer queue type is compatible with the printer emulation	A-8
Verifying the lp interface script is compatible with the printer connection method	A-10
Verifying remote system information for remshlp and network connections	A-11
Troubleshooting using data flow isolation techniques	A-12
Example 1	A-13
Example 2	A-17
Troubleshooting specific areas and programs	A-20
Terminfo and forms	A-20
STREAMS and printer devices	A-25
MiLAN FastPort device	A-33
Termservers	A-37
Print filters	A-44
Remote printing with remshlp	A-49
Printing with lpd and lpr	A-50

Appendix B – Determining tty lines for asynchronous ports

Port tty line numbers	B-1
Completing terminal controller worksheets	B-2
Determining the order of DG/UX device names for terminal line controllers	B-10
Learning the terminal line controller type and cluster controller type	B-11
Determining the tty line for terminal line controller ports	B-20
How the DG/UX system allocates tty lines	B-20

Appendix C – Printer Worksheets

Tables

Table

1-1	Valid printer types	1-5
1-2	Printer specific terminfo entries	1-9
2-1	Reference list of interface cables	2-15
2-2	Connecting asynchronous printers to the RS-232 system ports	2-16
2-3	Connecting asynchronous printers to a VDA cluster Box or VAC 16	2-16
2-4	Connecting parallel printers to the system parallel port	2-16
2-5	Connecting parallel printers to a VDA cluster box parallel port	2-16
3-1	Default attributes for local printers	3-2
4-1	Default attributes for remote printers connected via lpNet	4-6
5-1	Default attributes for remote remsh printers	5-6
5-2	Default attributes for remote remsh printers on AOS/VS II	5-14
6-1	Default attributes for networked printers	6-11
7-1	Default attributes for local printers	7-6
7-2	Default attributes for remote printers connected via lpNet	7-7
7-3	Default attributes for remote remsh printers	7-9
7-4	Default attributes for networked printers	7-10
8-1	LP print service menu and command summary	8-1
8-2	Content types	8-12
8-3	Default printer port characteristics	8-15
9-1	LP print service command reference	9-1
9-2	Default printer form attributes	9-5
9-3	Filter defaults	9-17
9-4	Filter option keywords	9-19
9-5	PostScript filters	9-39
9-6	Special function filters	9-40
9-7	stty options related to printers	9-49
9-8	terminfo items relevant to printers	9-50
9-9	Printer exit codes	9-56
A-1	Lp subsystem interface scripts	A-10
A-2	FastPort self-test problems	A-34

A-3 FastPort switch positions for Ethernet cables A-37

B-1 Default local boot device and system disk B-13

B-2 Number of tty Lines Allocated to Terminal Line Controllers B-21

B-3 tty Lines Allocated to Cluster Controller Addresses B-22

B-4 Lines Allocated to Systech Terminal Line Controllers and Cluster
Controllers B-24

Figures

Figure

2-1	VDC 128/255 cluster box (VDC/8P) pin number mapping	2-2
2-2	Asynchronous adapter cable	2-3
2-3	Model 15426 asynchronous 4-wire cable for VDA/VAC controllers	2-5
2-4	Asynchronous 4-wire software flow control cable with fault handling ...	2-6
2-5	Asynchronous 4-wire cable for CS TermServer 2000/2001	2-7
2-6	Asynchronous hardware cable for Datability VCP300 termserver	2-10
2-7	Asynchronous hardware cable for Datability VCP1000 termserver	2-10
2-8	Model 15426Exxx software/hardware busy async interface cable	2-17
2-9	Model 1340 EIA RS-232C interface cable	2-18
2-10	Model 15339E EIA RS-232C interface cable	2-18
2-11	Model 15307E RS-232C sw/hw busy I10 cable	2-19
2-12	Model 15275 TermServer/Modem interface cable	2-19
2-13	Asynchronous DTE 7-wire cable for ANNEX termservers	2-20
2-14	Asynchronous DTE 5-wire cable for ANNEX termservers	2-20
2-15	Asynchronous DTE 3-wire cable for ANNEX termservers	2-21
3-1	Configured devices in OS server system file	3-3
3-2	Configured devices in OS server client system file	3-4
8-1	Print server configuration	8-4
8-2	Network configuration	8-4
8-3	Methods of Connecting a Printer to a Computer	8-6
9-1	How LP processes print a request	9-47
9-2	Sample front banner page	9-64
9-3	Sample back banner page	9-64
A-1	Streams data transfer	A-27
A-2	VTC to termserver configuration	A-39
A-3	Ethernet controller to termserver configuration	A-42
B-1	Sample worksheet for terminal line controllers	B-4
B-2	Worksheet for terminal line controllers	B-5
B-3	Sample Worksheet for RS-232/422 Ports On Computer Unit	B-6
B-4	Worksheet for RS-232/422 Ports On Computer Unit	B-6
B-5	Sample Worksheet for a VAC/16 Controller	B-7
B-6	Worksheet for a VAC/16 Controller	B-7
B-7	Sample Worksheet for a VDA Host Adapter	B-8
B-8	Worksheet for a VDA Host Adapter	B-9

B-9 Automatically configured devices in the system configuration file **B-10**
B-10 First screen of system diagnostics **B-14**

1

Setting up printers and queues

This chapter discusses concepts on printing data stored on a computer system. The concepts are best understood through the tasks for configuring a printer to a host. The tasks are:

- Identifying physical printer characteristics
- Choosing the printer port
- Setting up queues in the DG/UX system
- Understanding document types

The following sections explain these tasks

Identifying physical printer characteristics

Printers can be grouped into two major categories: line and page printers. Line printers are usually dot matrix and band printers. Page printers are usually laser. Plotters are considered to be in the page printer category. Line and page printers are distinguished by several features. Among these features are paper handling, print style, and fault handling. Page printers are fed single sheets of paper from trays, print documents a page at a time, and respond to a hardware printer fault, such as a paper jam, by reprinting the page without intervention from the software printer scheduler. Line printers, on the other hand, are fed a continuous stream of perforated paper, print documents line by line, and depend on the software printer scheduler to resolve faults.

Printer emulations

Most printers support several printer emulations. For example, the DG Model 6618 printer emulates the following printers: MT660/690®, IBM Proprinter™, Epson FX™, and Printronix® P600. Each emulation supports a given set of commands, such as font selection. In addition, most printers support several commands that let you switch among the emulation modes, and to perform general setup procedures. These commands may be independent of any emulation mode.

When setting up your printer, you choose an emulation mode in which to run. The emulation mode may be selected manually from the printer's control panel, by the software application, or by the

line printer spooler. When setting up your printer to use with the line printer spooler, you must indicate the emulation mode. You choose an emulation mode in **sysadm** by selecting a printer type. See “Printer type” below.

Choosing the printer port

You must choose the method of connecting your printer to the host. You can connect a printer to the host through either the printer’s serial or parallel port. Most printers come equipped with a parallel port. Some printers have a serial port and others have both.

When printing large volumes of data, graphics files, or PostScript files, you should consider using a parallel port. Parallel connections send more bytes of data per second than a serial connection. Parallel cables, however, are shorter than serial cables, which means that parallel printers must be placed closer to the host.

For printing simple ASCII and small PostScript files, using a serial port is adequate. Serial connections send each data bit sequentially, which means that the flow of data is slower than a parallel connection. The advantage of the serial connection is the interface cable can be quite long. Thus, serial printers can be located much farther from the host. In addition, serial connections usually allow the printer to send more information, such as printer fault messages, back to the host than parallel connections.

In addition to serial and parallel ports, Data General also supports HP LaserJet 4 printers with network adapter cards and printers connected to a termserver or MiLAN print server device. These network devices allow you to place printers further from the host than parallel and serial ports.

The next two subsections briefly discuss how to set up printers using parallel and serial ports.

Setting up printers using a parallel port

The DG/UX system supports the following methods of setting up printers using parallel ports. They are:

- A direct connection to the parallel port **/dev/lp** on the host system
- A connection through the parallel port (**/dev/tty_n**) on a VDA cluster box (VDC/8p/16)
- A network connection to a parallel port on a MiLAN FastPort print server

Each of the above connections is set up differently in the DG/UX system. For a direct connection, use **sysadm** to add a local printer,

entering `/dev/lp` as the device name. Similarly, for a cluster box, use `sysadm` to add a local printer, but enter `/dev/tty` as the device name. For details about determining `tty` numbers, see Appendix B. See Chapter 3 for details on configuring these printers.

The parallel port `/dev/lp` and the parallel port device for the VDA cluster box are streams-based and support many `stty` options. These options include raw output for graphics printing and converting New Lines (NL) to carriage return and New Line pairs (CR/NL). Flow control for parallel devices is handled by the Ready/Busy protocol, which uses the busy signal in the interface cable to determine whether or not the printer is busy.

Setting up the parallel port on a MiLAN print server is different from the connections explained above. MiLAN print servers are accessed like other hosts on your network, using host names and Internet addresses. Connections through these print servers require a network connection type. Using `sysadm`, select the **Remote to networked printer device** connection type and the **fastport** interface when adding a printer connected to a MiLAN print server. See Chapter 6 for details on configuring printers attached to MiLAN print servers.

Setting up printers using a serial port

For serial ports, the DG/UX system lets you set up the following connections:

- A direct connection to a serial Duart port on the system board
- A connection through a serial port on a VDA cluster box (VDC/8p/16)
- A connection through a serial port on a VAC 16 cluster box
- A network connection to a serial port on a termserver
- A network connection to the serial port on a MiLAN FastPort print server

These connections are set up either as local or networked printers.

Setting up Duarts, VDA and VAC 16 cluster boxes

Use `sysadm` to add printers with serial connections. Add direct connections to a Duart and those through a VDA and VAC16 clusters boxes as local printers. For these connections, enter `/dev/tty` as the device name. For details about determining `tty` numbers, see Appendix B. See Chapter 3 for details on configuring these printers.

Printers connected to serial ports offer more flexibility in handling communication options than parallel connections. For example, serial connections allow you the option to use hardware flow control. Parallel connections do not provide the option of hardware flow control. Select an alternate form of flow control by setting the appropriate **stty** options when defining a printer. See Chapter 3 for details.

Setting up network connections

Network connections offer greater printer accessibility by allowing printers wherever there is an available Ethernet connection. These connections also allow multiple hosts using different operating systems to share printers.

When setting up multiple hosts to share a networked printer, you can have each host communicate directly to the networked printer or have a single host (the print server) communicate with the networked printer.

We recommend setting up a printer server. That way, the scheduler on the printer server controls print requests. The other method offers no global scheduling. Print requests are handled on a first-come-first-serve basis.

Setting up printers connected to termservers and the MiLAN print server present an added level of complexity. First, a serial connection must be established between the termserver or MiLAN print server and the printer. This connection offers flexibility in communications options, such as flow control. Users can choose whether to use software or hardware flow control. Second, a network connection must be established between the AViiON host and the termserver or MiLAN print server. Both termservers and MiLAN print servers communicate using the TCP/IP socket interface. They both have host names and Internet addresses.

Setting up queues in the DG/UX system

After you establish the port connection, you must set up the printer queues in the DG/UX system. This requires you to make decisions on how to configure some software and hardware parameters.

Software configurable parameters

The software configurable parameters initialize the operating system printer ports and the printer emulation modes to match the document being printed. Many printers can print several printer languages, such as ASCII, PostScript, and Epson FX. To print a

document with one of these languages, the user must prepare and deliver the file to the appropriate printer. The DG/UX system uses the following parameters to adjust the operating system: printer type, content type, and **stty** options.

Printer type

The printer type refers to the emulation mode on the printer. Printer type corresponds to the description of that printer in the **terminfo** database. It tells the LP print service the type of printer being added. The **terminfo** entries contain defined printer capabilities, such as how to set page size, line and character pitches, print quality, and printer initialization. The initialization string contains instructions for setting the initial state of the printer (for example, page size and pitches); resetting the printer; and, where applicable, setting the emulation mode.

The **terminfo** database files are located at `/usr/share/lib/terminfo/x/y`, where *x* is a single-letter name for the directory containing the printer type file *y*. For example, the **terminfo** file `/usr/share/lib/terminfo/e/epsonfx` is provided for the printer type “epsonfx.” The generic “printer-80” printer type can be used as a default for 80-column line printers.

Data General defines printer specific **terminfo** entries as *model number-emulation mode* (for example, `dg6617-epsonfx` or `dg6617-proprinter`). Printer specific **terminfo** entries initialize the emulation mode. Use **sysadm** to add printers and specify printer types. For details, see Chapters 3–6 on how to configure printers. Table 1–1 is a list of the valid printer types provided in the DG/UX system.

Table 1–1 Valid printer types

Printer Models	Description
Generic Printer Types	
printer	Generic line printer; 132 columns.
printer-80	Generic line printer; 80 columns.
lpr	Same as printer printer type.
lpr-80	Same as printer-80 printer type.
PostScript Printer Types	
dg6640T-PS	DG Model 6640T/6646 printer in PostScript (local serial).
dg6640T-PS-b	DG Model 6640T/6646 printer in PostScript (any non serial).
dg6640T-hp2	DG Model 6640T/6646 printer in PCL mode.

Continued

Printer Models	Description
dg6771-PS	DG Model 6771/6772 printer in PostScript® (local serial).
dg6771-PS-b	DG Model 6771/6772 printer in PostScript (any non serial).
dg6771-hp2	DG Model 6771/6772 printer in PCL mode.
PS	Generic PostScript printer for serial devices except termserver.
PS-b	Generic PostScript printer; batch mode for any parallel printer or serial termserver printer.
PS-br	PostScript printer; batch mode for any parallel printer or serial termserver printer; pages stack in reverse order.
PS-r	PostScript printer; for serial devices except termserver; pages stack in reverse order.
Epson Printer Types	
epsonfx	Epson FX® printer; 136 columns.
epsonfx-80	Epson FX printer; 80 columns.
oki320-epsonfx	Okidata Microline® 320/321 printer (Epsonfx mode); 80 columns.
oki321-epsonfx	Okidata a Microline 321 (Epsonfx mode); 136 columns.
oki321-epsonfx-80	Okidata Microline 320/321 printer (Epsonfx mode) ;80 columns.
dg6617-epsonfx	Data General 6617 line printer (Epsonfx mode); 132 columns.
dg6617-epsonfx-80	Data General 6617 line printer (Epsonfx mode); 80 columns.
dg6618-epsonfx	Data General 6618 line printer (Epsonfx mode); 132 columns.
dg6618-epsonfx-80	Data General 6618 line printer (Epsonfx mode); 80 columns.
dg6594-epsonfx	Data General 6594 line printer (Epsonfx mode); 136 columns.
dg6594-epsonfx-80	Data General 6594 line printer (Epsonfx mode); 80 columns.
dg6514-epson	Data General 6514 /6515 printer (Epson mode); 80 columns.
dg6515-epson	Data General 6515 printer (Epson mode); 136 columns.
dg6515-epson-80	Data General 6514/6515 printer (Epson mode); 80 columns.
epson800	Epson LQ-800/1000; 80 columns.
epson1000	Epson LQ-1000; 136 columns.
epson1000-80	Epson LQ-800/1000; 80 columns.
epson2500	Epson LQ-2500; low resolution; 136 columns.
epson2500-hi	Epson LQ-2500; high resolution; 136 columns.
epson2500-80	Epson LQ-2500; low resolution; 80 columns.
epson2500-hi80	Epson LQ-2500; high resolution; 80-columns.
Hewlett Packard Printer Types	
hp7550	Hewlett Packard 7550A.
hplaserjet	HP LaserJet® I.
hplaserjetII	HP LaserJet II.

Continued

Printer Models	Description
hplaserjetIII	Generic LaserJet 3 PCL mode.
hplaserjetIIISi	HP LaserJet IIISi™.
hplaserjetIII-PS	LaserJet 3 PostScript (local serial connection).
hplaserjetIII-PS-b	Laserjet 3 PostScript (any non local serial).
hplaserjetIII-PCL	Laserjet 3 PCL mode.
hplaserjetIIISi	Generic Laserjet 3Si PCL mode.
hplaserjetIIISi-PS	LaserJet 3Si PostScript (local serial connection).
hplaserjetIIISi-PS-b	LaserJet 3Si PostScript (any non local serial).
hplaserjetIIISi-PCL	Generic Laserjet 3Si PCL mode.
hplaserjet4	Generic LaserJet 4 PCL mode.
hplaserjet4-PS	LaserJet 4 PostScript (local serial connection).
hplaserjet4-PS-b	Laserjet 4 PostScript (any non local serial).
hplaserjet4-PCL	Laserjet 4 PCL mode.
IBM Printer Types	
ibmgraphics	IBM Graphics printer.
ibmcolor	IBM Color printer; 5:6 aspect ratio.
ibmcolor-1x1	IBM Color printer; 1:1 aspect ratio.
ibmxi	IBM Proprinter® XL; 5:6 aspect ratio; low resolution; 136 columns.
ibmxi-80	IBM Proprinter XL; 5:6 aspect ratio; low resolution; 80 columns.
ibmxi-1xi	IBM Proprinter XL in AGM; 1:1 aspect ratio; low resolution; 136 columns.
ibmxi-1xi-hi	IBM Proprinter XL; in AGM; 1:1 aspect ratio; high resolution; 136 columns.
ibmxi-1xi-80	IBM Proprinter XL in AGM; 1:1 aspect ratio; low resolution; 80 columns.
ibm-1xi-hi80	IBM Proprinter XL in AGM; 1:1 aspect ratio; high resolution; 80 columns.
proprinter	IBM Proprinter; 136 columns.
proprinter-80	IBM Proprinter; 80 columns.
proprinterXL	IBM Proprinter XL; 136 columns.
proprinterII	IBM Proprinter II; 80 columns.
oki321-proprinter	Okidata Microline® 321 printer (Proprinter mode); 136 columns.
oki321-proprinter-80	Okidata Microline 320/321 printer (Proprinter mode); 80 columns.
oki320-proprinter	Okidata Microline 320/321 printer (Proprinter mode); 80 columns.
dg6594-proprinter	Data General 6594 line printer (Proprinter mode); 136 columns.
dg6594-proprinter-80	Data General 6594 line printer (Proprinter mode); 80 columns.
dg6618-proprinter	Data General 6618 line printer (Proprinter mode); 132 columns.

Continued

Printer Models	Description
dg6618–proprinter–80	Data General 6618 line printer (Proprinter mode); 80 columns.
dg6617–proprinter	Data General 6617 line printer (Proprinter mode); 132 columns.
dg6617–proprinter–80	Data General 6617 line printer (Proprinter mode); 80 columns.
Other Printer Types	
citoh	Citoh 8510 printer.
daisy	Daisy brand printer.
qume	Qume Sprint 11.
la100	DEC™ LA100 Printer.
ln03	DEC LN03 Laser Printer.

Content type

The content (or input) type tells the LP print service what types of files can be printed on each printer. Most printers can print at least two types of files: the type supported by the printer emulation (for example, PostScript) and **simple** (an ASCII file, which is the default for most printers).

Some printers can accept several types of files, such as LaserJet™, Epson FX, and PostScript. However, we do not recommend specifying more than one content type when adding a printer. You can use **sysadm** to specify content types. For details, see chapters 3–6 on how to configure printers.

stty options

The characteristics for the operating system printer port, which are set through **sysadm**, should match those set on the printer. **stty** options allow you to adjust those characteristics. They let you control data conversion and define communication with the printer. **stty** options let you do the following:

- Convert tabs into spaces.
- Convert New Lines into carriage return and New Line pairs.
- Set the baud rate.
- Set bits per byte.
- Set parity.

For details, see chapters 3–6 on how to configure printers.

Hardware configurable parameters

When setting up queues for printers connected to term servers and MiLAN print servers, the port parameters must be set up to communicate with the printer. Parameters that must be set include:

- Flow control.
- Communications input (serial or parallel connection).
- Bin selection.
- Timeout values for multiple ports.
- Emulation modes.

The parameters that apply and their values depend on the particular printer being added, how you want data handled, and how you configure the printer.

When setting up hardware parameters, you must decide on the connection-related parameters (for example, baud rate and parity) and parameters related to page layout and print style (for example, margins and font style). These parameters set the stage for printing, but do not really manipulate the text. You must also decide on parameters that manipulate data as it is printing. These parameters perform functions such as changing tabs to spaces. For these parameters, you must decide where you want changes to the data to occur. If you want the printer to remain constant, for example, then all data manipulation would occur on the host through DG/UX system parameters, not on the printer. For details, see Chapter 6, "Configuring Networked Printers."

Adding printers with multiple emulation modes

The DG/UX system provides printer specific terminfo entries (also called printer types) that allow you to use one printer in multiple emulation modes. These terminfo entries let you use a printer in both PostScript (PS) and Printer Control Language (PCL) emulations concurrently. Table 1-2 shows the printer specific entries that allow mode switching.

Table 1-2 Printer specific terminfo entries

Printer Models	Description
PostScript Printer Types	
dg6640T-PS	DG Model 6640T/6646T printer in PostScript (local serial).
dg6640T-PS-b	DG Model 6640T/6646T printer in PostScript (any non serial).
dg6640T-hp2	DG Model 6640T/6646T printer in PCL mode.
dg6646T-PS	DG Model 6646T printer in PostScript (local serial).
dg6646T-PS-b	DG Model 6646T printer in PostScript (any non serial).
dg6646T-hp2	DG Model 6646T printer in PCL mode.
dg6771-PS	DG Model 6771 printer in PostScript (local serial).

Continued

Printer Models	Description
dg6771-PS-b	DG Model 6771 printer in PostScript (any non serial).
dg6771-hp2	DG Model 6771 printer in PCL mode.
dg6772-PS	DG Model 6772 printer in PostScript (local serial).
dg6772-PS-b	DG Model 6772 printer in PostScript (any non serial).
dg6772-hp2	DG Model 6772 printer in PCL mode.
Hewlett Packard Printer Types	
hplaserjetIII-PS	LaserJet 3 PostScript (local serial connection)
hplaserjetIII-PS-b	Laserjet 3 PostScript (any non local serial)
hplaserjetIII-PCL	Laserjet 3 PCL mode
hplaserjetIIISi-PS	LaserJet 3Si PostScript (local serial connection)
hplaserjetIIISi-PS-b	LaserJet 3Si PostScript (any non local serial)
hplaserjetIIISi-PCL	LaserJet 3Si PCL mode
hplaserjet4-PS	LaserJet 4 PostScript (local serial connection)
hplaserjet4-PS-b	Laserjet 4 PostScript (any non local serial)
hplaserjet4-PCL	Laserjet 4 PCL mode

To use automatic mode switching, use **sysadm** to add two printers (with different names) that send files to the same physical printer. Add one printer, supplying one of the above printer specific PS entries (for example, **hplaserjet4-PS**). Add the second printer, supplying the corresponding PCL entry (for example, **hplaserjet4-PCL**). Be sure to define both printers to use the same physical printer, (for example, **/dev/lp**). For details on how to add printers with **sysadm**, see Chapters 3–6.

Understanding document types

Document type refers to the kind of file you can print on the printer. Document types work with printer types to ensure that the file sent to the printer is in the form the printer expects. Each printer type has a unique command set. The document type is the format of this command set.

When setting up printers in the DG/UX system, the document type must match the printer type. The LP subsystem performs any necessary processing to prepare a document for the printer. Some software applications, however, produce documents that are formatted for a particular document type. These documents contain commands appropriate for a particular printer and do not require processing from the LP subsystem. Such documents are in a *print-ready* state. For these documents, be sure to supply the **stty**

option **-opost** to prevent output processing when configuring the printer .

Simple ASCII and PostScript are two of the more popular formats. Simple ASCII, which prints on almost all printers, includes the upper- and lower-case letters, numbers, and symbols entered from standard keyboards. It also includes the following non-printable control characters: Horizontal Tabs, Form Feed, Carriage Return, and New Line.

PostScript is a formatting language with its own commands and codes. The following example shows a file in the PostScript language.

```
%!PS-Adobe-2.0
%%Version: 3.15
%%DocumentFonts: (atend)
%%Pages: (atend)
%%EndComments
/vmstat save def%
% Copyright (C) Data General Corporation, 1984 - 1991
% All Rights Reserved.
% Licensed Material-Property of Data General Corporation.
% This software is made available solely pursuant to the
% terms of a DGC license agreement which governs its use.
%
%      <@(#) postprint.ps,v      5.1.1.1>

%ident  "@(#)lp:filter/postscript/postscript/postprint.ps
1.1"
%
% Version 3.15 prologue for ASCII files.
%

/#copies 1 store
/aspectratio 1 def
/font /Courier def
/formsperpage 1 def
/landscape false def
% DG - Added dynamic point size and margin calculation
/linespp 66 def
/magnification 1 def
% DG - Added dynamic point size and margin calculation
/margin -1 def
/orientation 0 def
% DG - Added dynamic point size and margin calculation
/pointsize 0 def
/spacing .10 def
/xoffset 0 def
/yoffset 0 def
```

End of Chapter

2

Connecting printers

This chapter discusses the concepts involved in connecting printers to hosts and network devices. It covers Duart ports, VDA/VAC controllers, term servers, VTC boards, MiLAN FastPort print servers and interface cables.

Interconnectivity concepts

The Electronic Industries Association (EIA) defined RS-232C, which is a standard interface for connecting asynchronous computers and peripherals. This standard specifies two interface connections. One is Data Terminal Equipment (DTE) and the other is Data Circuit-terminating Equipment (DCE). Initially, DTE specified connections for computers, terminals, and serial printers. DCE, on the other hand, specified connections only for modems. DCE now specifies connections for some computers as well.

Because computers can have either a DCE or a DTE connection, you must know the appropriate connection when ordering cables. Equipment based on the DTE connection sends data on pin 2 and receives data on pin 3. Equipment based on the DCE connection operates just the opposite, sending data on pin 3 and receiving data on pin 2. Cables for the respective connections are not interchangeable.

In RS-232C, the standard DTE connection contains 25 wires that pass control signals and data from a computer to a peripheral. The list below shows pin functions for each wire.

1 Frame Ground	10 Negative dc Test Voltage	19 Sec. Request To Send
2 Transmitted Data (TXD)	11 Unassigned	20 Data Terminal Ready (DTR)
3 Received Data	12 Sec. Data Carrier Detect	21 Signal Quality Detect
4 Request to Send (RTS)	13 Sec. Clear to Send	22 Ring Indicator (RI)
5 Clear to Send (CTS)	14 Sec. Transmitted Data	23 Data Rate Select
6 Data Set Ready (DSR)	15 Transmitter Clock	24 Ext. Transmitter Clock
7 Signal Ground	16 Sec. Received Clock	25 Busy
8 Data Carrier Detect (DCD)	17 Receiver Clock	
9 Positive dc Test Voltage	18 Receiver Dibit Clock	

Figure 2–1 shows how the pins are mapped to signals for the Systech cluster controller, which is a DCE connection. The less than symbol (<) indicates that the signal is being received by the cluster controller, while the greater than symbol (>) indicates that the signal is being transmitted from the cluster controller.

Pin Number	Signal Name
1	Protective ground
2	RXD <
3	TXD >
4	CTS <
5	RTS >
6	DTR >
7	Signal ground
8	DCD >
20	DSR <

Figure 2–1 VDC 128/255 cluster box (VDC/8P) pin number mapping

Output from term servers, VDC boards, and VAC 16 controllers is suitable for a DCE connection.

A Duart connection, which is a DTE connection, typically uses a 9-pin connector cable to pass control signals and data from a computer to a peripheral. For Duart connections, we recommend you use the 005-38420 asynchronous adapter cable shown in Figure 2-2. This cable converts the Duart connection to a DCE connection. If you choose not to use the adapter cable, follow the instructions for connecting to Duart ports in the next section.

DB 25 (Female)		DB 9 (Female)
RCD 2	—————	2 RCD
TXD 3	—————	3 TXD
CTS 4	—————	8 CTS
RTS 5	—————	7 RTS
DTR 6	—————	4 DTR
GND 7	—————	6 DTR
DCD 8	—————	1 DCD
DSR 20	—————	6 DCR
RI 22	—————	9RI

Figure 2-2 Asynchronous adapter cable

Connecting to hosts

This section discusses connecting a printer directly to system ports and asynchronous controllers for serial and parallel communication. The method required to connect a printer to a host depends on the port device on the host and printer. Both the printer and host must support the type of connection you choose—serial or parallel.

Connecting to a Duart port

When the connection port on the AViON is a Duart, the connection to the printer must be serial (RS-232C). Duart ports do not support parallel connections. Choosing a cable depends on the model of the AViON and the type of flow control required. Use the information in Table 2-1 to determine the appropriate cables.

The DG/UX system, by default, supports software flow control, using the signals XON and XOFF. To use hardware flow control, you must disable software flow control, enable hardware flow control, and rewire the connection between host and printer according to the scheme in your building.

When you add or modify a printer, use the **stty** options to disable software flow control and enable hardware flow control. Disable software flow control by specifying the options **-ixon** and **-ixoff** using either the **sysadm** menus or the **lpadmin** command. Enable hardware flow control by specifying the option **ctsxon**. See Chapter 3 for details.

Connecting to VDA/VAC controllers

A printer connection to a VDA or VAC controller can be parallel or serial (RS-232C). The cable required depends on the particular controller you use, the type of connection, and the type of flow control you want. Tables 2-3 and 2-4 list the cables necessary for connecting serial and parallel printers, respectively, to VDA and VAC controllers.

When connecting parallel printers, flow control usually is not an issue. Flow control in these connections is defined by the Centronics interface. Parallel connections transfer data using a strobe signal that indicates the data lines are open from the host. The host checks the Ready and Busy signals to determine if the printer is ready to accept data.

The DG/UX system supports software flow control by default, using the signals XON and XOFF. To use another method of flow control, you must disable the default flow control. In particular, if you want to use hardware flow control with a serial connection, you must enable hardware flow control and rewire the connection between host and printer according to the scheme in your building.

If you decide to rewire the cabling between the controller and printer, use the pinout diagram in Figure 2-3 for a 4-wire asynchronous hardware and software cable that could be used for hardware flow control. The model number for this cable is 15426E nn , where nn is the length of the cable.

IMPORTANT: The following pinout diagram represents a cable for a printer that uses DTR when its buffer is full. This cable does not work for printers that do not use DTR when their buffer is full.

Printer DB 25 (Male)	Controller DB 25 (Male)
TXD 2	2 RCD
RXD 3	3 TXD
GND 7	7 GND
DTR 20	4 CTS
RTS 4	6 DTR
CTS 5	8 DCD
DSR 6	20 DSR
DCD 8	

Figure 2-3 Model 15426 asynchronous 4-wire cable for VDA/VAC controllers

Connecting to network devices

This section discusses connecting a printer to Termprinter and Datability termservers, VTC boards, and MiLAN print servers.

Connecting to Term servers

Term servers require two connecting procedures:

- Establish the network connection between the AViiON system and the Term server
- Establish the serial connection between the printer and the Term server

The network connection between the AViiON system and the Term server is an Ethernet connection with a TCP/IP socket interface. Refer to the hardware manual for your particular term server for details on how to establish a network connection.

The next two sections explain how to establish a serial connection between the printer and term server. They present examples of how to set up CS 2000/2100 series Term Servers and Datability termservers, respectively.

The DG/UX system supports the following interface scripts for termserver: **termprinter_3com3.3**, **termprinter_tcp**, **termprinter_telnet**, **dg_annex**, and **datability**. While the scripts support different types of termserver, all of the termserver support serial printers. Because of the serial connection between termserver and printer, you can choose to use hardware rather than software flow control. The type of cable required depends on whether you choose software or hardware flow control. For software flow control, you can refer to Table 2-1 for the correct cable, but if you want fault handling, rewire your connection as shown in Figure 2-4. For hardware flow control, you must disable software flow control, enable hardware flow control, and rewire the connection between termserver and printer. In most cases, refer to Table 2-1 for the correct cable. If the cable listed in Table 2-1 does not work, use the wiring diagram in Figure 2-5 to build a cable.

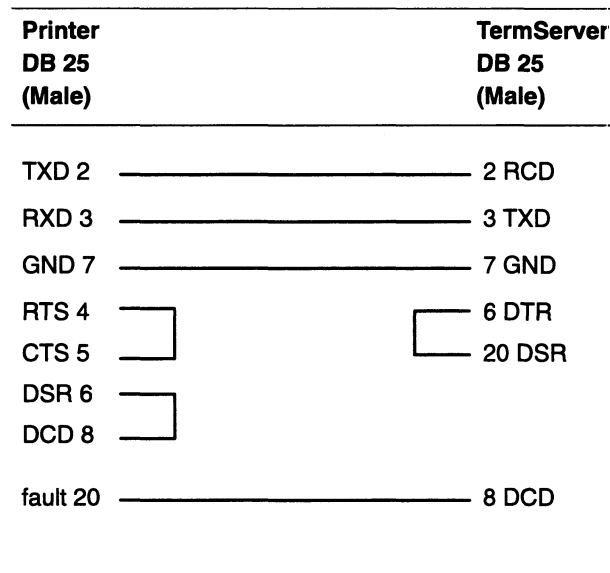


Figure 2-4 Asynchronous 4-wire software flow control cable with fault handling

IMPORTANT: The fault wire may be located on a pin other than pin 20. See the manual for your printer hardware to determine how your printer is wired.

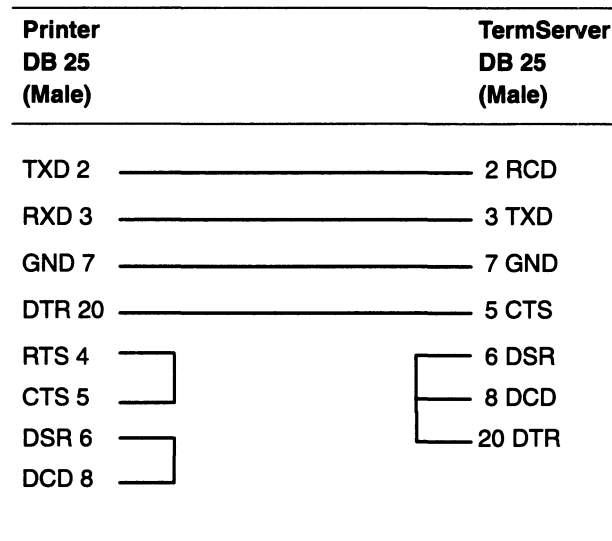


Figure 2-5 Asynchronous 4-wire cable for CS TermServer 2000/2001

Setting up a CS 2000/2100 Series TermServer

To configure the TermServer port, set the parameters listed in this section and other parameters as necessary. Set the **Baud**, **FlowControlTo**, and **FlowControlFrom** parameters to match the printer (use “**setd parameter = value**” for permissible values).

To use hardware flow control, design a special cable so that the printer’s DTR signal is connected to the termserver’s CTS signal. Set the **UseDTRin** parameter to “Ignore,” and set the **FlowControlTo** and **FlowControlFrom** parameters to “CTS_RTS.”

Suppose you want to set up for a printer with the following characteristics on serial port number 1:

- 9600 baud rate
- 8 bit data
- no parity
- software flow control with XON/XOFF

From the TermServer, follow these steps:

1. Enable the ability to issue configuration commands on the TermServer by setting the **PRi** parameter.

```
CS/200T> set pri=gnm ↵
password: ↵
```

- Use the **setd** command to set the device parameter.

```
cs/200t# setd (!) device = (host, paper, def)
```

- Set the remaining parameters as follows:

```
cs/200t# setd (!) Baud =9600
cs/200t# setd (!) DataBits = 8
cs/200t# setd (!) DUplex = full
cs/200t# setd (!) LinePRotocol = Asynchronous
cs/200t# setd (!) PARity = None
cs/200t# setd (!) StopBits = 1
cs/200t# setd (!) UseDCDout = AlwaysAssert,NoToggle
cs/200t# setd (!) UseDTRin = Ignore
cs/200t# setd (!) FlowControlTo = Xon_Xoff
cs/200t# setd (!) FlowControlFrom = Xon_Xoff
cs/200t# setd (!) Mode = Transparent
```

- Issue the command **sh (!) dp** to verify the parameter settings.

```
.....Port Transmission and VTP Characteristics.....
Bufferize = 82      DeVice = ( Terminal, Glass, DeFault )
InterAction = ( Verbose, Echo, MacroEcho, MacroBreak, Broadcast ON; NoLFInsert)
Init Macro= ""      MaxSessions = 1      PRivilege = User

.....Port Physical Characterstics .....
BAud = 9600          BSPad = None          CRPad = None          FFPad = None
LFPad = None         TabPad = None          DataBits = 8          DUplex = Full
LinePRotocol = Asynchronous          PARity = ( None )    StopBits = 1
UseDCDout = ( AlwaysAssert, NoToggle )  UseDTRin = Ignore

.....Session Transmission and VTP Characteristics .....
BreakAction = ( EscDTM)                BReakChar = Disabled
DIscnectAction = None                   DataForward = None  ECHOData = OFF
ECHOMask = ( AlphaNum, CR, Term, Punct )  ECMchar = ^^
NetAsci = UseLF          FlowControlFrom = ( Xon_Xoff )
FlowControlTo = ( Xon_OFF )          FlushVC = OFF      IdleTimer = 2
LongBReakAction = IIgnore          LFInsertion = None  Mode = Transparent
XmitBinary = OFF      XOFF = ^S          XON = ^Q

.....Session & User Interface Editing parameters.....
ERASE = ^H
Localediting = ( NoDataEditing, CmdEditing )          ReprintLine = ^R
VERBatim = ^V          WordERASE = ^W
```

- If using the Data General Model 664x and 677x series laser printers and software flow control, set the parameter, **FlowControlFrom**, to **None**. This is required for sending 8-bit data (PCL format) to these laser printers in HP LaserJet mode.

Setting up a Datability TermServer

You can choose between two methods of establishing access to specific ports on a Datability TermServer. The first method is to

create a unique IP address for a port, associate that address with a specific port, and set the port characteristics. The second method is to create a unique IP address for a port, associate that address with multiple ports, and set the port characteristics. Execute the commands below on the TermServer to accomplish these tasks. Before executing the commands, make sure the `Local>` prompt appears on the console and you issue the command `SET PRIV` to set the super user privilege. The default password is "SYSTEM."

This example defines a printer port named P1 on termserver port 8, with an IP address of 192.41.217.209 and a speed of 9600 baud.

Enter the following commands for each printer port on the termserver. Note that the **Change Address** command adds new addresses, but does not change the address of the unit. You can define up to 63 additional addresses. To display the port parameters, enter the command **SHOW PORT 8**.

```
Local> CHANGE ADDRESS 192.41.217.209
```

```
Local> CHANGE PORT 8 AUTOBAUD DISABLE ACCESS REMOTE SPEED 9600
```

```
Local> CHANGE PORT 8 IO_FLUSH DISABLE
```

```
Local> CHANGE SERVICE P1 PORT 8 IP 192.41.217.209 TCP PORT 23
```

```
Local> CHANGE SERVICE P1 PORT 8 TELNET DISABLE LPR ENABLE
```

A typical configuration for connecting a printer to a Datability TermServer would have the following parameters:

```
Character Size: 8  
Flowcontrol: XON -XOFF (software)  
Parity: none  
Stop Bits: 1  
Access: Remote  
Input Speed: 9600
```

When rewiring your connections for hardware flow control, use the pin out diagrams in Figures 2-6 and 2-7.

TermServer RJ45 (Male)		Printer DB 25 (Male)
RTS 1	_____	5 CTS
DTR 2	_____	6 DSR
TXD+ 3	_____	3 RXD
GND 4	_____	7 GND
RXD- 5	_____	7 GND
RXD+ 6	_____	2 TX
DSR 7	_____	20 DTR
CTS 8	_____	4 RTS

Figure 2-6 Asynchronous hardware cable for Datability VCP300 termserver

TermServer RJ12 (Male)		Printer DB 25 (Male)
DTR 1	_____	6 DSR
TXD+ 2	_____	3 RXD
GND 3	_____	7 GND
RXD- 4	_____	7 GND
RXD+ 5	_____	2 TX
DSR 6	_____	20 DTR
DTR 20	_____	5 CTS
		4 RTS
		5 CTS

Figure 2-7 Asynchronous hardware cable for Datability VCP1000 termserver

Connecting to VTC boards

The VTC is a VME controller for an AViiON system with an Ethernet interface that connects to a LAN. Printers connect directly to the termserver, not to the VTC. When setting up the printer definitions for a printer to be accessed through the VTC, use the same hardware options as those for connecting a printer to a termserver. Use many of the same software options as those for a

standard local serial printer. The DG/UX kernel will, however, ignore certain software options, such as baud rate. The DG/UX system treats the VTC as a local printer, using the streams interface, the `/dev/tty n` port, and the standard interface script.

To connect a printer to a termserver through a VTC, follow these steps:

1. Install the VTC hardware. Each VTC in the system must be included in the kernel. The VTC uses the same device names as the VAC/16 and VDA/255 devices. Each VTC must appear on a separate line in the system configuration file. The DG/UX system assigns 256 ttys to each VTC that is configured into the system. The tty assignments are based on the order of devices in the system configuration file. All 256 ttys respond to the same Internet Address unless you specify a specific Internet Address for particular ttys. For more information, see *Managing the DG/UX™ System*.
2. Add entries to the file `/etc/tcload/vtc.addrs` that specify the VTC's Internet address, broadcast address, netmask, and routing information the board should use. This file should also contain entries for any ttys that need specific Internet addresses. For details, see the manual page `vtc.addrs(4M)`.
3. Install the termserver hardware and software. This involves configuring serial ports for the printers you attach. See the termserver documentation for details.
4. Determine the correct cable to connect the printer to the TermServer. See Table 2–1 for a list of cables.
5. One option is to set up the port as a termserver with a permanent telnet connection to the IP address as defined in the `vtc.addrs` file.
6. Set up the host to print to the termserver by adding the printer to the DG/UX system as a local device. Use `sysadm` to accomplish these tasks, following the instructions in Chapter 3.

Setting up a printer accessed through the VTC

This example shows the required procedures on the host and the Termserver for setting up a printer to be accessed through the VTC. This example uses the following:

- An AViiON system named **bird** that contains a LAN interface (VLC hken0) and VTC board
- A termserver named **dizzy** that is connected to the LAN
- A terminal named **miles** that is connected to the LAN
- A printer connected to the termserver **dizzy**

Follow the procedure below:

1. Add a line to the file `/etc/tload/vtc.addrs` that includes the device name (`/dev/tty40`) and the Internet address (**128.999.11.2**) as shown below.

```
#
# VTC Board Spec
#
# Internet      Broadcast
# Addr          Addr          Netmask      Routing
#
/dev/async/syac@60(60000000) 128.999.3.166 128.999.3.255 255.255.255.00 default
# The following is an entry for the printer.
/dev/tty40                128.999.11.2
```

2. Issue the `tload` command so the VTC will recognize the IP address associated with a tty entry.

```
# /usr/sbin/tload -a
```

3. Enable the ability to issue configuration commands on the termserver, `dizzy`, with this command:

```
CS/200T> set pri=gnm ↵
password: ↵
```

4. Set the device parameter by entering:

```
cs/200t# setd (!1) device = (host, glass, def) ↵
```

5. Use `setd` as shown above to set the remaining parameters:

```
Baud =9600
DataBits = 8
DUplex = full
LinePRotocol = Asynchronous
PARity = None
StopBits = 1
UseDCDout = AlwaysAssert,NoToggle
UseDTRin = Ignore
FlowControlTo = Xon_Xoff
FlowControlFrom = Xon_Xoff
Mode = Transparent
```

Because you set port (!1) to `host`, you can use the Permanent Virtual Circuit (PVC) parameter to initiate a connection from the termserver, `dizzy`, to the VTC. This parameter forces `dizzy` to issue a connection request to IP 128.999.11.2. If the connection fails, `dizzy` tries continuously. Because 128.999.11.2 is associated with `/dev/tty40` in the file `/etc/tload/vtc.addrs`, when the connection is made, `dizzy`'s port 1 is considered by the AViON system, `bird`, as `/dev/tty40`.

6. Set the PVC parameter to 128.999.11.2 (the printer).

```
cs200#> setd (1) pvc = 128.999.11.2 ↵
```

7. Force port 1 to read its default parameters.

```
cs200#> listen (1) ↵
```

8. From the terminal manager, **miles**, connect to the termserver, **dizzy**.

```
cs100#> remote dizzy
```

9. Configure port 1 on the terminal manager, **miles**, using the **setd** command as in steps 4 and 5 above.
10. Configure the printer on the AViiON system, **bird**, using **sysadm**. Be sure to set the **stty** option **-hupcl**. This option maintains the connection from the TermServer, **dizzy**, to the VTC. Using the **-hupcl** option prevents printer sharing. For details on how to configure local printers, see Chapter 3.

Connecting to MiLAN FastPort print servers

Like TermServers, MiLAN print servers require the following two connecting procedures:

- Establish the network connection between the host via the Ethernet and the MiLAN print server
- Establish a connection between the printer and MiLAN print server.

The network connection between the Ethernet and the MiLAN print server can be established using 10BaseT, 10Base2, or 10Base5 Ethernet media. Refer to the *FastPort 3000 and 3100 Network Print Server User's Guide* for details.

MiLAN print servers have both an RS-232C serial and a Centronics parallel port. The cable required to connect the MiLAN print server to the printer depends on the type of connection (serial or parallel). See Table 2-1 for a list of cables.

Following the instructions in the *FastPort 3000 and 3100 Network Print Server User's Guide*, connect the printer to a MiLAN print server. You must:

1. Install the MiLAN print server on the network and connect printers to it.
2. Set up a **rarpd** server with the MiLAN print server's Ethernet and IP address, or enter the IP address into the MiLAN print server's nonvolatile RAM, using a **telnet** or serial connection.

3. Enter the MiLAN print server's host name into the host tables of all spooling hosts.
4. Then you are ready to add the printers to the DG/UX system using **sysadm** or **lpadmin**. See Chapter 6.

Interface cables

This section lists the cables used to connect your asynchronous printers to your AViiON system.

The interface cable you use to connect your printer to your system depends on your AViiON system type and your port/controller type. You can determine the port/controller type from where the port is located: the AViiON system Duart port, a Systech VAC/16 controller, or VDA controller. When you have determined the port/controller type, use Table 2-1 to find the appropriate cable.

A termserver is a network device, requiring cabling different from system and controller ports. For more specific cabling information, see the setup guide that came with your system. Table 2-1 lists cable models, lengths, and part numbers for cables that are available.

Table 2-1 Reference list of interface cables

Cable model	Length	Part number	System*	Purpose
10235	5 ft.	005-23915	W3	parallel printer to system parallel port
1338	25 ft.	005-33775	ALL	async printer to modem or termserver (DCE)
1339	25 ft.	005-33783	W4	async device to RS-422 system port (50 to 500 ft. extension cables are available, models 1339-A, B, C, and D.)
1340-S	5 ft.	005-13325	W1	async printer (SW busy only) to RS-232 system port
1340-T	15 ft.	005-33703	W1	async printer (SW busy only) to RS-232 system port
1340	25 ft.	005-33788	W1	async printer (SW busy only) to RS-232 system port
1340-A	50 ft.	005-33776	W1	async printer (SW busy only) to RS-232 system port
15275E025	25 ft.	005-33764	ALL	async modem to termserver (DCE)
15293E030	30 ft.	005-33762	W3	parallel printer to system port. Also used on a parallel printer to a MiLAN FastPort print server.
15307E025	25 ft.	005-32727	W1	async printer (HW/SW busy) to RS-232 system port
15339E025	10 ft.	005-34255	S	async printer (HW busy only) to modem or termserver (DCE).
15339E015	15 ft.	005-34992	S	
15339E025	25 ft.	005-34993	S	
15426E006	6 ft.	005-040782	W2	async printers (HW/SW busy) to RS-232 system port. Use with adapter cable (005-38420). Replaces Model 15340.
15426E015	15 ft.	005-040787	W2	
15426E025	25 ft.	005-040788	W2	
15426E006	6 ft.	005-040782		Used on all systems that support async printers to a VDA or VAC controller port. Replaces Model 15340.
15426E015	15 ft.	005-040787		
15426E025	25 ft.	005-040788		
15345E015	15 ft.	005-37910	S	parallel printer to system port
15345E025	25 ft.	005-37911	S	parallel printer to system port
40263E006	6 ft.	119-003303	ALL	MiLAN FastPort print server to async printer port
	0.5 ft.	005-38420	W2	adapter cable for RS-232 system ports

* AviiON System Key

W1 = AV 100-300, 400, 3200, 4000, or 4100 series systems only

W2 = AV 500, 530, 4600, or 5500 series systems only

W3 = AV 300, 400, 530, 3200, 4000, 4100, or 4600 series only

W4 = AV 100-300 series systems only

W5 = AV 7400, 8400, 8500, 9500

S = AV 5200, 5500, 6200, 7000, and 8000 series systems only

All = All AViiON series systems listed above

Interface cables listed by device-type and purpose

Tables 2–2 through 2–5 list the cables above according to how they are used. Refer to Table 2–1 for part numbers and available cable lengths.

IMPORTANT: For cable models in the format 12345xxx, xxx indicates the cable length.

Table 2–2 Connecting asynchronous printers to the RS–232 system ports

System	Printer model	Cable model
AV 100, 200, 300, 400, 3200, 4000, 4100	6514, 6515, 6617, 6618, 10692, 10693, 10696, 18782, 18783	15307Exxx with software busy
	6647, 6648, 6594, 6640, 6646, 6779, 6771, 6772, 6773	15307Exxx with software busy
AV 530, 4600		15426Exxx with adapter 005–38420

Table 2–3 Connecting asynchronous printers to a VDA cluster Box or VAC 16

System	Cable model
any system	15426xxx

Table 2–4 Connecting parallel printers to the system parallel port

System	Cable model
AV 300, 400, 530, 3200, 4000, 4100, 4300, 4600	15293E030 or 10235
AV 5200, 6200, 7000, 8000	15345Exxx

Table 2–5 Connecting parallel printers to a VDA cluster box parallel port

System	Cable model
any system	15345Exxx

Cabling diagrams for serial printers

This section provides diagrams of the cables listed in the previous section. These diagrams show how signals move through the cables between your printer and your AViiON system. It also shows wiring diagrams for cables to Annex-XL and Annex-3 TermServers.

IMPORTANT: For cable models in the format 12345xxx, xxx indicates the cable length.

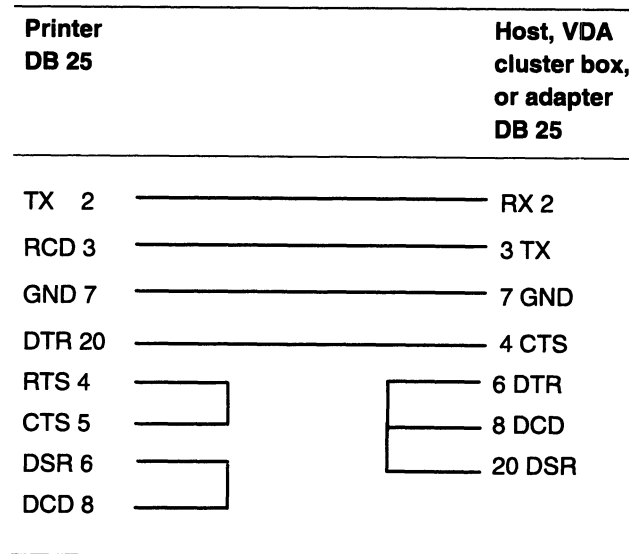


Figure 2-8 Model 15426Exxx software/hardware busy async interface cable

When using the cable above on AV 530/4600 systems, you must use a 9-to-25 pin adapter cable, Data General part number 005-38420. Figure 2-2 shows the cabling diagram for this adapter.

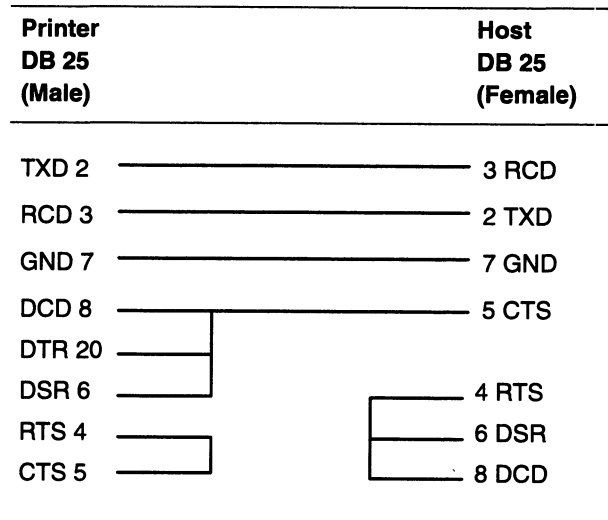


Figure 2-9 Model 1340 EIA RS-232C interface cable

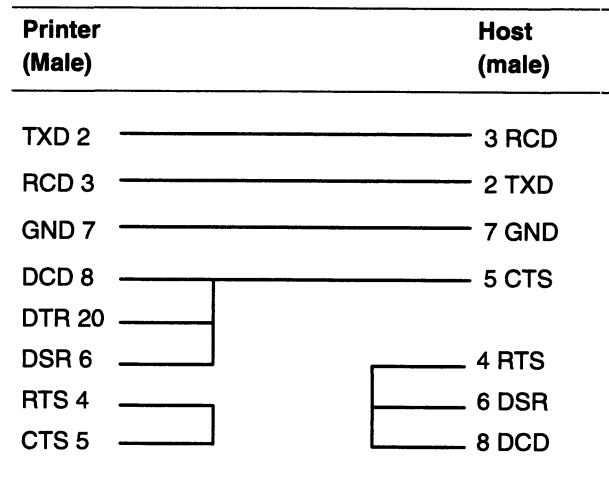


Figure 2-10 Model 15339E EIA RS-232C interface cable

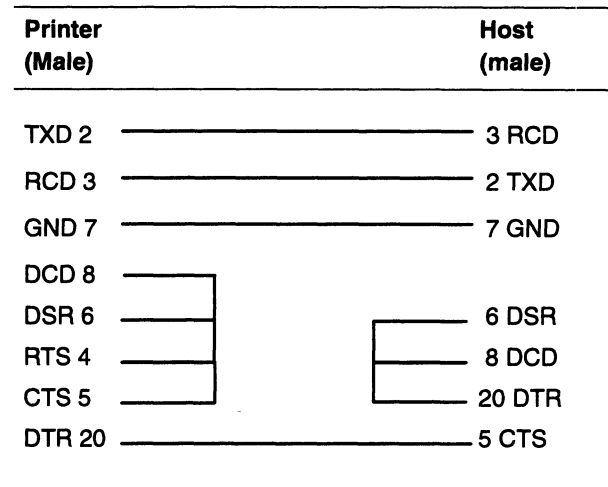


Figure 2-11 Model 15307E RS-232C sw/hw busy I10 cable

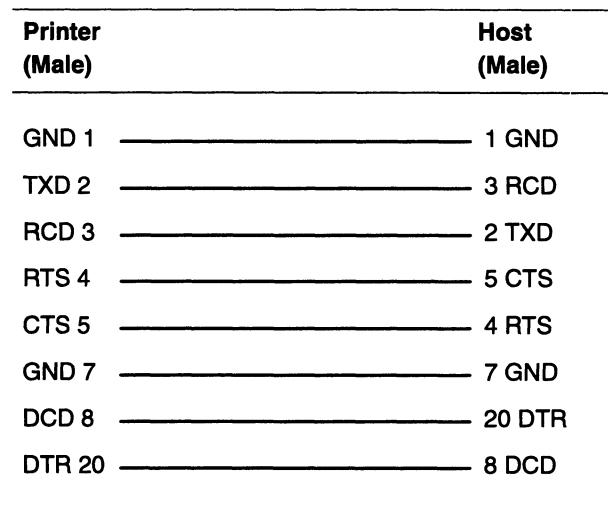


Figure 2-12 Model 15275 TermServer/Modem interface cable

Setting up Annex-XL and Annex-3 TermServers

When rewiring your connections for the ANNEX-XL and ANNEX-3 series termservers, use the pin out diagrams below. Figure 2-13 shows a pin out diagram for a cable that supports hardware and software flow control and all control signals.

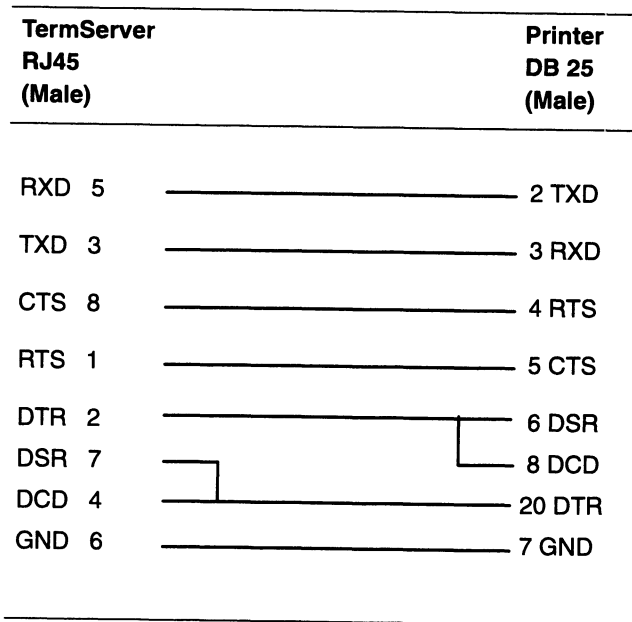


Figure 2-13 Asynchronous DTE 7-wire cable for ANNEX termserver

Figure 2-14 shows a pin out for a cable that supports only software flow control and all control signals.

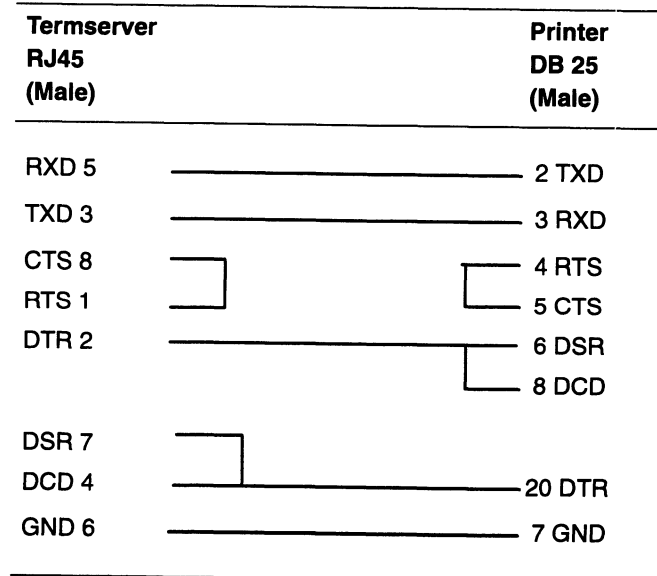


Figure 2-14 Asynchronous DTE 5-wire cable for ANNEX termserver

Figure 2-15 shows the pin out diagram for a special cable that supports only software flow control and no control signals.

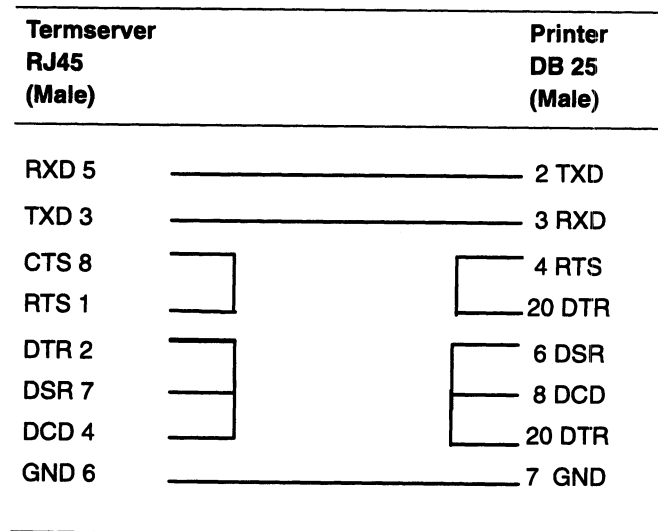


Figure 2-15 Asynchronous DTE 3-wire cable for ANNEX termserver

Parallel printer cables

Cable Model 15293E030 or 10235 is a standard “PC/AT” Centronics parallel cable, with a “Champ” (Amphenol) connector on the peripheral end and a 25DB connector on the host end.

Cable Model 15345Exxx is a standard Centronics parallel cable, with a “Champ” (Amphenol) connector on both ends.

End of Chapter

3

Configuring local printers

This chapter shows how to configure local printers. Local printers can be connected to a local port (**/dev/lp**), to a Duart port (**/dev/tty**), or to a Syac controller (**/dev/tty**). This chapter describes how to configure local printers and gives examples for configuring these printers:

- Adding a line printer connected to **/dev/lp**.
- Adding a PostScript printer connected to **/dev/lp**.
- Adding an Epsonfx printer connected to a Duart port
- Adding a PostScript printer connected to a Syac port
- Adding a PostScript printer connected to **/dev/lp** on a Trusted DG/UX system

The **sysadm** program offers two methods of adding printers: a quick add and a full add feature. The quick add feature assumes you will accept the default attributes for a given connection type. The full add feature prompts you for all attributes, allowing you to modify the default values. “Adding a line printer connected to **/dev/lp**” illustrates using the quick add feature and “Adding a PostScript printer connected to **/dev/lp**” illustrates how to change default values using the full add feature.

When adding local printers with the Quick Add feature, **sysadm** supplies default values for the less common printer attributes (Table 3-1).

Table 3–1 Default attributes for local printers

Printer attribute	Default value
Interface script	standard
Stty options	clocal
Input options	nobanner
Device is also a login	no
Fault recovery	continue
Alert messages	mail
Alert interval	0
Users to allow	all
Users to deny	N/A
Forms to allow	none
Forms to deny	N/A
Enable printer	yes
Accept printer	yes

IMPORTANT: Be sure to complete hardware installation before following the procedures in the following sections.

Adding printers connected to /dev/lp

The **/dev/lp** port is a local parallel port. This port is a streams device. This lets you use **stty** options (that you set when defining a printer) to control data conversion and post processing. As with other parallel connections, the Centronics interface controls low level communications, such as flow control, parity, and baud rate. The options you set depend on the mode in which your printer is set and the type of data you send to it.

To add a printer that uses **/dev/lp**, you must

1. gather printer information,
 2. ensure that the kernel recognizes the local printer port, and
 3. add the printer using **sysadm**.
- Gather and record the following information. Use the forms in Appendix C.

printer name A printer name can contain these characters: **a-z A-Z 0-9 _**. The name should begin with a letter or digit and be no more than 14 characters long.

connection type	Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. For /dev/lp , the connection type is Local .
device file	Specifies the pathname of the port to which you connected the printer. The pathname is /dev/lp .
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> . For example, the terminfo file /usr/lib/terminfo/e/epsonfx is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. See Table 1-1 for a list of valid printer types.

- Make sure the **lp()** line controller is listed in the system configuration file.

The default name of the OS server or stand-alone computer's system file is **system.hostname**, which is located at **/usr/src/uts/aviion/Build**. The default name of the OS client's system file is **system.diskless**, which is also located at **/usr/src/uts/aviion/Build**.

Use **more** to view the system file for devices configured automatically when you booted the installer system during installation. If your hostname is **hawk**, you could view the file as follows:

```
# cd /usr/src/uts/aviion/Build ↵
# more system.hawk ↵
```

Figure 3-1 shows an excerpt of a typical OS server's system file.

```
sd(insc(0),*) ## all SCSI disks on Integrated SCSI adapter
st(insc(0),*) ## all SCSI tapes on Integrated SCSI adapter
duart()      ## integrated Duart terminal line controller
lp()         ## integrated parallel line printer controller
syac()       ## Systech terminal line controller
syac(1)      ## second Systech terminal line controller
syac(2)      ## third Systech terminal line controller
```

Figure 3-1 Configured devices in OS server system file

Figure 3–2 shows an excerpt of a typical OS client's system file.

```

Typical Hardware devices for an OS Client:
#
# Note that your system can have an lp() controller or a second
# duart() but not both! You'll have to determine which situation
# applies.
#
#
# kbd( )          ## Workstation keyboard
# grfx( )        ## Workstation graphics display
# duart(0)       ## Dual-line terminal controller (number 0)
# duart(1)       ## Dual-line terminal controller (number 1)
# inen( )        ## Integrated Ethernet controller
# lp( )          ## Integrated parallel line printer controller

```

Figure 3–2 Configured devices in OS server client system file

If the appropriate terminal line controller does not appear in the system file, add the appropriate entry, rebuild, and boot the kernel. For information on rebuilding your kernel, see *Managing the DG/UX™ System*.

Adding a line printer connected to /dev/lp

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

```
Printer name: laser ↵
```

3. Supply the connection type you want to use. Accept the default, Local.

```
Connection type: [Local /dev/tty or /dev/lp] ↵
```

4. Accept the Quick Add feature.

```
Quick Add using default values? [yes] ↵
```

5. Accept the default printer type.

```
Printer type: [printer-80] ↵
```

6. Accept the default input type.

Input types: [simple] ↵

The default, `simple`, refers to an ASCII file.

7. Enter the pathname of the physical printer device.

Device: **/dev/lp** ↵

8. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Draft laser printer on hallway 21.** ↵

The note accompanies the printer name whenever you list the status of added printers.

9. Confirm your choices:

OK to perform operation? [yes] ↵

After answering this question, your local printer is added and ready for operation.

Adding a PostScript printer connected to /dev/lp

This example adds a printer that accepts PostScript files. This example uses the full add procedure. For a list of the default values of the printer attributes, see Table 1-1.

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

Printer name: **laser2** ↵

3. Supply the connection type you want to use. Accept the default, Local.

Connection type: [Local /dev/tty or /dev/lp] ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Enter the PostScript printer type. For a list of valid printer types, enter the help (?) command or see Table 1-1.

Printer type: [printer-80] **PS -b** ↵

6. Enter **PS-b** (PostScript) for the input type. Input types represent the final formats that the printer can accept and affect how your print request is filtered.

Input types: [simple] **PS** ↵

7. Accept the standard interface script.

Interface script: [standard] ↵

8. Enter the pathname of the physical printer device.

Device: **/dev/lp** ↵

9. Enter the stty option **-opost** to turn off output processing.

Stty options: [clocal] **-opost**

10. Accept the defaults for the following attributes:

Print options: [nobanner] ↵
Device is also a login terminal: [no] ↵
Fault recovery: [continue] ↵
Alert messages: [mail] ↵
Alert interval: [0] ↵
Users to allow: [all] ↵
Users to deny: ↵
Forms to allow: [none] ↵
Forms to deny: ↵
Enable? [yes] ↵
Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Laser printer on doc hallway.** ↵

12. Confirm your choices:

OK to perform operation? [yes] ↵

laser2 has been added.
Printer laser2 has been enabled.
Printer laser2 has been set to accept requests.

Adding printers connected to a Duart

The Duart port is a serial port. When connecting printers to these ports, you have the option to set low-level communications, such as flow control, baud rate, and parity. The default **stty** settings and standard interface script in the DG/UX system are set to use software flow control with the signals XON and XOFF. If, for example, you have chosen to use hardware flow control, you must turn off software flow control and turn on hardware flow control, using the **stty** options. You can make these changes in **sysadm**.

To add a printer to a Duart port, you must gather printer information, ensure that the kernel recognizes the local printer port, determine the port's tty line number, and add the printer using the **sysadm**.

- ▶ Gather and record the following information on your printer. Use the forms in Appendix C.

printer name	A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication and determines the queries when adding or modifying a printer. For a Duart port, the connection type is Local .
device file	The port to which you connected the printer. For example, /dev/tty01 is the device file for a serial printer on the port assigned tty01 .
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> . For a list of valid printer types, see Table 1-1.

- ▶ Make sure the **duart()** line controller is listed in the system configuration file.

The default name of the OS server or stand-alone computer's system file is **system.hostname**, which is located at **/usr/src/uts/aviion/Build**. The default name of the OS client's system file is **system.diskless**, which is also located at **/usr/src/uts/aviion/Build**.

Use **more** to view the system file for devices configured automatically when you booted the installer system during installation. If your hostname is **hawk**, you could view the file as follows:

```
# cd /usr/src/uts/aviion/Build ↵
# more system.hawk ↵
```

See Figures 3–1 and 3–2 for excerpts from a typical OS server’s and a typical OS client’s system file.

If the appropriate terminal line controller does not appear in the system file, add the appropriate entry, rebuild, and boot the kernel. For information on rebuilding your kernel, see *Managing the DG/UX™ System*.

► Determine a port’s tty line number

The DG/UX system automatically assigns a tty line number to each serial port in the hardware configuration when the system boots. A tty line number takes the form:

ttix

where *x* is a decimal number containing at least two digits that are sequentially assigned. For example, **tty00** refers to the first port, **tty01** the second port, and so on.

For serial printers, a file is created bearing the name of the tty line number in the **/dev** directory each time the system boots. For details on determining tty lines for asynchronous ports, see Appendix B.

Adding an Epsonfx printer connected to a Duart

This example adds a printer that accepts Epsonfx files and uses hardware flow control. It uses **sysadm**’s full add feature. When using the full add feature, **sysadm** prompts you for all attributes, allowing you to modify the default values. For a list of the default values of the printer attributes, see Table 1–1.

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

Printer name: **epson-laser** ↵

3. Supply the connection type you want to use. Accept the default, Local.

Connection type: [Local /dev/tty or /dev/lp] ↵

4. Do not accept the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Enter the Epsonfx printer type. For a list of valid printer types, enter the help (?) command or see Table 1-1.

Printer type: [printer-80] **epsonfx** ↵

6. Enter **epsonfx** for the input type. Input types represent the final formats that the printer can accept and affect how your print request is filtered.

Input types: [simple] **epsonfx** ↵

7. Accept the standard interface script.

Interface script: [standard] ↵

8. Enter the pathname of the physical printer device.

Device: **/dev/tty02** ↵

9. In this case, you need to turn off output processing, turn off software flow control, and turn on CTS hardware flow control. Enter the stty options **-opost, -ixon, -ixoff, and ctsxon**.

Stty options: [clocal] **-opost -ixon -ixoff ctsxon**

10. Accept the defaults for the following attributes:

Print options: [nobanner] ↵

Device is also a login terminal: [no] ↵

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Epson printer on doc hallway.** ↵

12. Confirm your choices:

OK to perform operation? [yes] ↵

epson-laser has been added.
Printer epson-laser has been enabled.
Printer epson-laser has been set to accept requests.

Your local printer is ready for operation.

Adding printers connected to a Syac

Syac (VDC/VDA and VAC16) controllers can support both parallel and serial printer connections. However, to the DG/UX system, a Syac is similar to a Duart. It is a local device and is attached to a **tty** port. When connecting printers to these ports, you have the option to set low-level communications, such as flow control, baud rate, and parity. The default **stty** settings and standard interface script in the DG/UX system are set to use software flow control with the signals XON and XOFF. If, for example, you have chosen to use hardware flow control, you must turn off software flow control and turn on hardware flow control, using the **stty** options. You can make these changes in **sysadm**.

To add a printer to a Syac controller, you must gather printer information, ensure that the kernel recognizes the local printer port, determine the port's tty line number, and add the printer using the **sysadm**.

- ▶ Gather and record the following information on your printer. Use the forms in Appendix C.

printer name	A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication and determines the queries when adding or modifying a printer. For a Syac controller, the connection type is Local .
device file	The port to which you connected the printer. For example, /dev/tty01 is the device file for a serial printer on the port assigned tty01 .

printer type Specifies the **terminfo** database file, which describes your printer's capabilities. The **terminfo** database files are located at **/usr/lib/terminfo/x/y**, where *x* is a single-letter name for the directory containing the printer type file *y*. For a list of valid printer types, see Table 1-1.

- ▶ Make sure the **syac()** line controller is listed in the system configuration file.

The default name of the OS server or stand-alone computer's system file is **system.hostname**, which is located at **/usr/src/uts/aviion/Build**. The default name of the OS client's system file is **system.diskless**, which is also located at **/usr/src/uts/aviion/Build**.

Use **more** to view the system file for devices configured automatically when you booted the installer system during installation. If your hostname is **hawk**, you could view the file as follows:

```
# cd /usr/src/uts/aviion/Build ↵
# more system.hawk ↵
```

See Figures 3-1 and 3-2 for excerpts from a typical OS server's and a typical OS client's system file.

If the appropriate terminal line controller does not appear in the system file, add the appropriate entry, rebuild, and boot the kernel. For information on rebuilding your kernel, see *Managing the DG/UX™ System*.

- ▶ Determine a port's tty line number

The DG/UX system automatically assigns a tty line number to each attached printer serial port in the hardware configuration when the system boots. A tty line number takes the form:

ttyx

where *x* is a decimal number containing at least two digits that are sequentially assigned. For example, **tty00** refers to the first port, **tty01** the second port, and so on.

For serial printers, a file is created bearing the name of the tty line number in the **/dev** directory each time the system boots. For details on determining tty lines for asynchronous ports, see Appendix B.

Adding a PostScript printer connected to a Syac

This example adds a printer that accepts PostScript files and uses hardware flow control. It uses **sysadm**'s full add feature. When using the full add feature, **sysadm** prompts you for all attributes, allowing you to modify the default values. For a list of the default values of the printer attributes, see Table 3-1.

1. Follow this path through **sysadm**:

Device → Printer → Devices → Add

sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

Printer name: **laser_ps1** ↵

3. Supply the connection type you want to use. Accept the default, Local.

Connection type: [Local /dev/tty or /dev/lp] ↵

4. Do not accept the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Enter the PostScript printer type. For a list of valid printer types, enter the help (?) command or see Table 1-1.

Printer type: [printer-80] **PS** ↵

6. Enter PS (PostScript) for the input type. Input types represent the final formats that the printer can accept and affect how your print request is filtered.

Input types: [simple] **PS** ↵

7. Accept the standard interface script.

Interface script: [standard] ↵

8. Enter the pathname of the physical printer device.

Device: **/dev/tty04** ↵

9. In this case, you need to turn off output processing, prevent the SIGHUP signal, turn off software flow control, and turn on CTS hardware flow control. Enter the stty options **-opost**, **-clocal**, **-ixon**, **-ixoff**, and **ctsxon**.

Stty options: [clocal] **-opost -clocal -ixon -ixoff ctsxon**

10. Accept the defaults for the following attributes:

```

Print options: [nobanner] ↵
Device is also a login terminal: [no] ↵
Fault recovery: [continue] ↵
Alert messages: [mail] ↵
Alert interval: [0] ↵
Users to allow: [all] ↵
Users to deny: ↵
Forms to allow: [none] ↵
Forms to deny: ↵
Enable? [yes] ↵
Accept? [yes] ↵

```

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

```

Printer Description: Production printer in lab 14. ↵

```

12. Confirm your choices:

```

OK to perform operation? [yes] ↵

```

```

laser_ps1 has been added.
Printer laser_ps1 has been enabled.
Printer laser_ps1 has been set to accept requests.

```

Your local printer is ready for operation.

Adding local printers on a Trusted DG/UX system

IMPORTANT: Data General offers B1 and C2 Trusted DG/UX systems. Both are optional and must be purchased in addition to the DG/UX system. This section applies only if you have purchased one of the Trusted DG/UX systems.

When adding printers on a Trusted DG/UX system, the procedure and concepts are the same as presented in the previous sections of this chapter. For B1 security, however, you must answer two additional prompts. The prompts are `MAC Range Lo:` and `MAC Range Hi:`. `MAC Range Lo:` indicates the lowest MAC label at which the printer can accept jobs. `MAC Range Hi:` is the highest MAC label at which the printer can accept jobs. For detailed information on MAC labels, see *Managing Security on the Trusted DG/UX™ System*.

To add a PostScript printer to `/dev/lp`, for example, follow the same procedures documented in “Adding printers connected to `/dev/lp`.” That is, you do the following:

1. gather printer information,
2. ensure that the kernel recognizes the local printer port, and
3. add the printer using `sysadm`.

When you are ready to add the printer using `sysadm`, follow the procedures in the next section.

WARNING: If you overloaded this release on a previous security release, you must add your printer definitions again. If you overloaded this release on a regular DG/UX system, you must modify the print queues to include a MAC range. Do this by performing the modify operation in the the Device -> Printer -> Devices menu of `sysadm`. You will be prompted for the low and high values of the MAC range. Users cannot print to those queues until you modify them.

Adding a PostScript printer connected to `/dev/lp`

This example adds a printer that accepts PostScript files. This example uses the full add procedure. For a list of the default values of the printer attributes, see Table 1-1.

1. Follow this path through `sysadm`:

```
Device -> Printer -> Devices -> Add
```

`sysadm` guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

```
Printer name: laser2 ↵
```

3. Supply the connection type you want to use. Accept the default, Local.

```
Connection type: [Local /dev/tty or /dev/lp] ↵
```

4. Refuse the Quick Add feature.

```
Quick Add using default values? [yes] no ↵
```

5. Enter the PostScript printer type. For a list of valid printer types, enter the help (?) command or see Table 1-1.

Printer type: [printer-80] **PS -b** ↵

6. Enter **PS** (PostScript) for the input type. Input types represent the final formats that the printer can accept and affect how your print request is filtered.

Input types: [simple] **PS** ↵

7. Accept the standard interface script.

Interface script: [standard] ↵

8. Enter the pathname of the physical printer device.

Device: **/dev/lp** ↵

9. Enter the stty option **-opost** to turn off output processing.

Stty options: [clocal] **-opost**

10. Accept the defaults for the following attributes:

Print options: [nobanner] ↵
 Device is also a login terminal: [no] ↵
 Fault recovery: [continue] ↵
 Alert messages: [mail] ↵
 Alert interval: [0] ↵
 Users to allow: [all] ↵
 Users to deny: ↵
 Forms to allow: [none] ↵
 Forms to deny: ↵
 Enable? [yes] ↵
 Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Laser printer on doc hallway.** ↵

12. Supply the lowest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Lo: [IMPL_LO] ↵

13. Supply the highest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Hi: [ACR_HI] ↵

14. Confirm your choices:

OK to perform operation? [yes] ↵

laser2 has been added.

Printer laser2 has been enabled.

Printer laser2 has been set to accept requests.

End of Chapter

4

Configuring remote printers connected via lpNet and LPD

This chapter explains how to configure printers that are connected to a remote host running DG/UX Release 5.4 or greater and the lpNet daemon. It also describes how to configure printers connected to a remote host that supports the Line Printer Daemon Protocol (LPD), also referred to as the Berkeley print spooler.

This chapter contains the following examples of how to configure printers:

- Adding a remote PostScript printer using lpNet
- Adding a PostScript printer connected to a Novell system
- Adding a remote line printer connected to a SunOS system
- Adding a remote PostScript printer using lpNet on a Trusted DG/UX System

When configuring a remote printer on your AViiON system, you must determine what operating system is running on the remote host. Depending on the operating system, the print scheduler is either UNIX System V or Berkeley's BSD. For example, if the remote system is running DG/UX system 5.4 or greater with **lpsched**, the print scheduler is based on UNIX System V.

When adding remote printers, make sure that the printer is physically connected to the remote system you specify. The DG/UX system does not allow you to send print requests through one remote system to reach a printer connected to yet another remote system.

The DG/UX system does not allow you to define multiple remote queues for the same remote device. For example, let's suppose that you want to use a printer, **lp1**, that is attached to a remote system called **hawk**. On your local system, **trax**, you can set up a queue (for example, **hawk_lp1**) to send files to the remote machine. Once you have established the queue **hawk_lp1**, you cannot define another queue on **trax** to print to **lp1** on the remote system **hawk**.

When adding printers from another vendor's system, particularly BSD printers that run the LPD print spooler, be aware that the procedures for connecting to remote systems are explained from a BSD point of view. AViiON systems running the DG/UX system, by default, use the UNIX System V **lpsched** program. Thus, when adding a printer from another operating system, such as NetWare or SunOS, make sure you do the following:

- Check the vendor's documentation to determine whether the system uses the lpNet or the LPD program. Though Novell does not use the LPD program, NetWare can be set up to communicate with another LPD scheduler.
- Add remote printers connected to remote systems that run the LPD printer daemon. Use the following sections, which present examples using the **sysadm** program, as a guide.

IMPORTANT: Be sure to complete hardware installation before performing the procedures in the following sections.

Adding remote printers connected to DG/UX systems and LPD systems

This section describes how to add remote printers connected to DG/UX System 5.4 Release 2.01 (and later). Some of the procedures here also apply to BSD systems.

Allowing host access to remote printer resources

Allowing host access means that your local computer can use a printer on a remote machine. To allow host access, you must first register the remote host on your local system, then register your local system on the remote host. To register a remote host on your local system, follow the path below through **sysadm**. To register your system on a remote DG/UX system, log in to the remote system and follow the same path through **sysadm**. For SunOS systems, add an entry for the DG/UX system in the file **/etc/host.equiv** or **/etc/hosts.lpd**. For NetWare systems, add an entry for the DG/UX system to the trusted hosts list.

1. Device -> Printer -> Systems -> Add

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Specify the names of systems to share printer services. Enter one or more hostnames separated by spaces.

System name: **chrysalis harpo anapest** ↵

3. Choose the printer scheduler type used by the remote system. The scheduler type indicates the network protocol the remote system uses. Valid choices are **s5** for UNIX System V (the default) or **bsd** for BSD Line Printer Daemon (LPD) Protocol, SunOS, and Novell Netware.

Scheduler Type: [s5] ↵

4. Specify the number of minutes that the connection with the remote system may remain idle before terminating.

Connection Timeout: [10] ↵

The default 10 specifies that a connection can remain idle for ten minutes without terminating. To specify that the connection terminate as soon as it becomes idle, type 0.

5. Specify the number of minutes to wait following a disconnection of service with a remote system before attempting to re-establish the connection. Enter 2 for better response.

Retry Period: [10] **2** ↵

Type **no** if you do not want the print service to retry the connection. In this case, it retries when it receives a job request to send to the remote system. Type **0** if you want the print service to retry the connection immediately after any disconnection.

6. Enter a descriptive note about the remote system, such as the name of its system administrator, a phone number, or location.

Comment : **System administrator is Jon Harker** ↵

Do not use single quotation marks in the note. The note appears when you list information about the remote printer.

7. Confirm your entries:

OK to perform operation? [yes] ↵

"chrysalis" has been added.

"harpo" has been added.

"anapest" has been added.

After allowing host access, you can add remote printers on your local system. This section gives an example of how to add a printer in line printer mode. To add a remote printer, you must gather remote host and printer information to provide **sysadm** and add the printer using **sysadm**.

- ▶ Gather and record the following information. Use the forms in Appendix C.

remote host name Name of the remote DG/UX system.

remote printer name Name of the printer on the remote DG/UX system.

local printer name Name you give the remote printer on your local system. A printer name can contain these characters: **a-z A-Z 0-9 _**. The name should begin with a letter or digit and be no more than 14 characters long.

connection type Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. To communicate with another DG/UX system through lpNet, the connection type is **Remote lpNet to DG/UX 5.4. system**.

printer type Specifies the **terminfo** database file, which describes your printer's capabilities. The **terminfo** database files are located at **/usr/lib/terminfo/x/y**, where *x* is a single-letter name for the directory containing the printer type file *y*.

For example, the **terminfo** file **/usr/lib/terminfo/e/epsonfx** is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. See Table 1-1 lists valid printer types.

Adding a remote PostScript printer using lpNet

The following example uses this configuration:

Remote Host: chrysalis
Remote Printer name: dps
local printer name: psdoc
connection type: Remote lpNet to DG/UX 5.4. system
printer type: PS-b

1. Follow this path through sysadm:

Device -> Printer -> Devices -> Add

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default or enter a new value.

2. Enter a name for the remote printer. This can be any name you choose.

Printer name: **psdoc** ↵

3. Supply the connection type. Enter **Remote lpNet to DG/UX 5.4 system** to indicate a DG/UX 5.4 system.

Connection type: [Local /dev/tty or /dev/lp] **Remote
lpNet to DG/UX 5.4 system** ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type.

Printer type: [printer-80] **PS-b**

If you do not know the type of printer on the remote system, accept the default value (printer-80).

6. Enter the name of the remote host to which the printer is connected. (You must have added this host through Device -> Printer -> Systems -> Add as explained in the previous section, "Allowing host access to remote printer resources.")

Remote Host: **chrysalis** ↵

7. Enter the name of the remote printer. This is the name of the printer as defined on the remote host.

Remote Printer Name: **dps** ↵

8. Accept the default values for the following printer attributes:

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

9. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Production laser printer on doc hallway.**

The note accompanies the printer name when you list the status of the printer.

10. Confirm your entries. A warning appears from sysadm reminding you that your system must be added to the remote system and the remote printer must exist on the remote system.

```
OK to perform operation? [yes] ↵
You will not be able to print successfully unless
the system administrator for the remote system has
added your system via Device->Printer->Systems->Add
operation from sysadm. The remote printer will also
need to exist currently on the remote system.
psdoc has been added.
Printer psdoc has been enabled.
Printer psdoc has been set to accept requests.
```

Default attributes for remote printers connected via lpNet

When adding remote printers connected via **lpNet** with the quick add option, **sysadm** supplies default values for the less common printer attributes (Table 4-1).

Table 4-1 Default attributes for remote printers connected via lpNet

Printer attribute	Default value
Printer type	printer-80
Input types	simple
Users to allow	all
Users to deny	N/A
Forms to allow	none
Forms to deny	N/A
Enable printer	yes
Accept printer	yes
Interface script	N/A
Stty options	N/A
Print options	N/A
Device is a login	N/A
Fault recovery	N/A
Alert messages	N/A
Alert interval	N/A

Adding printers connected to Novell NetWare systems

Novell NetWare® systems do not use the LPD program, but offer an interface to systems using the LPD protocol. Though the DG/UX system employs the UNIX System V method of printing, which uses

the lpNet program, the **sysadm** program offers an interface to some systems using LPD protocol. Adding printers connected to Novell NetWare systems requires you to perform some tasks on both the Novell and the DG/UX systems. This section provides the procedures for adding printers that are connected to Novell systems.

Configuring Novell systems for remote printing

Before performing the tasks in this section, be sure that the Novell system has the following:

- The TCP/IP software package loaded.
- The NetWare NFS or FLeX/IP software package loaded.
- A local printer connected and running properly.
- The following NetWare Loadable Modules (NLM) loaded:

PLPD.NLM This software collects remote print jobs and distributes them in NetWare print queues.

PLPDCFG.NLM This utility configures the UNIX-to-NetWare (or DG/UX-to-NetWare) print server.

FILTER.NLM The software converts print jobs between ASCII and PostScript formats.

FLTRLIB.NLM This library supports FILTER.NLM.

Refer to the Novell NetWare documentation set for information on establishing the above environment.

When the programs above are loaded properly, you must perform the tasks listed below on the NetWare system. This list gives only an overview. For detailed instructions, see the manual *Netware NFS Supervisor's Guide*.

1. Add the DG/UX system hostname to the **/etc/hosts** file on your system. If your system uses the Network Information Service (NIS), add the DG/UX system hostname to the **/etc/hosts** file on the NIS master server.
2. Make print queues on the NetWare system available to the DG/UX host. Use the PLPDCFG utility to accomplish this task. PLPDCFG displays a menu from which you choose `Select Print Queues` for use by LPD. Follow the menus to choose available queues or add new ones to make available to the DG/UX system.
3. Add the DG/UX system to the list of trusted hosts on the NetWare machine. Use the PLPDCFG utility for this task. From the main menu, choose `Select Trusted Hosts`. Follow the menus to select the name of the DG/UX system.

4. Map usernames from the DG/UX system to usernames on the NetWare system. Use the PLPDCFG utility for this task. From the main menu, choose `Select Username Mapping Mode`. Follow the menus to choose a method mapping usernames. We recommend the single account mode. With this mode, every print job is mapped to a single NetWare username.

IMPORTANT: The DG/UX system uses the UNIX System V implementation of remote printing. Ignore procedures in the Novell NetWare documentation that cover connections with remote systems based on the BSD implementation. These procedures include making entries in the `printcap` file.

Configuring the DG/UX system for printing to NetWare systems

When you have completed the procedures on the NetWare system, you must perform three major tasks before printing on the NetWare system:

- Allow host access to the NetWare machine.
- Verify that LpNet is configured to use LPD (port 515) services.
- Add a printer on the NetWare machine to your client system.

► Allow host access to the NetWare machine

Allowing host access means that your local computer can use printers on the NetWare machine. To allow host access, you must add the NetWare machine's hostname and IP address to the `/etc/hosts` file and register the NetWare machine on your local system.

You can add the NetWare machine's hostname and IP address using `sysadm`. If you are using fully qualified domain names (`host.domain`), make sure the NetWare hostname is fully qualified. Add the name to NIS or DNS if applicable. To add the NetWare machine's hostname and IP address to the `hosts` file, do the following:

1. Follow this path through `sysadm`:

Networking -> TCP/IP -> Databases -> Hosts -> Add

`Sysadm` guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Supply the name of the host to allow remote access.

Host Name: **netware_serv** ↵

3. Supply the Internet address of the host to allow access.

Internet Address: **128:999.10.3** ↵

4. **sysadm** prompts you to supply an alias for the host. An alias is an option; it is not required.

Alias List: ↵

5. Press **Enter** to accept the values that you have entered.

OK to perform operation? [yes] ↵

netware_serv has been added.

The remote system **netware_serv** can access the host using the remote shell (**rsh** or **remsh**) or the remote login (**rlogin**). Refer to the **rsh(1)**, **remsh(1)**, and **rlogin(1)** manual pages for more information.

Next, you need to register the NetWare machine.

1. Follow this path through **sysadm**:

Device -> Printer -> Systems -> Add

2. Specify the names of systems to share printer services.

System name: **netware_serv** ↵

3. Choose the printer scheduler type used by the remote system. The scheduler type indicates the network protocol the remote system uses. Valid choices are **s5** for UNIX System V (the default) or **bsd** for BSD Line Printer Daemon (LPD) Protocol, SunOS, and Novell Netware.

Scheduler Type: [s5] **bsd** ↵

4. Accept the default value for the number of minutes that the connection with the remote system may remain idle before terminating.

Connection Timeout: [10] ↵

5. For better response, enter **2** as the number of minutes to wait following a disconnection of service with a remote system before attempting to re-establish the connection.

Retry Period: [10] **2** ↵

6. Enter a descriptive note about the remote system, such as the name of its system administrator, a phone number, or location.

3. If the standard LPD Berkeley daemon is running, modify the file `/etc/dgux.params`, removing the word "start" from the `lpd_START` parameter.

4. Stop and start the tcp port monitor using **sysadm** as follows:

```
Device -> Port -> Port Monitor -> Stop
```

Specify the tcp port monitor.

```
Port monitor tags:tcp ↵
```

Select **sysadm** option 7 to restart the port monitor. Specify the tcp port monitor.

```
Port monitor tags:tcp ↵
```

The tcp port monitor has been restarted.

After completing the procedures for allowing host access, you can add remote printers on your local system. To add a remote printer, you need to gather remote host and printer information to provide **sysadm** and then add the printer using **sysadm**.

- ▶ Gather and record the following information. Use the forms in Appendix C.

remote host name	Name of the machine running NetWare that the printer is connected to.
local printer name	Name you give the remote printer on your local machine. A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. To communicate with a NetWare system through lpNet, the connection type is Remote lpNet to lpd system .
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> . See Table 1-1 for a list of valid printer types.

Adding a remote PostScript printer connected to a NetWare system

The following example uses this configuration:

```
remote host:          netware_serv
remote printer name:  psdoc
local printer name:   papadoc
connection type:      Remote lpNet to lpd system
printer type:         PS-b
```

Follow these steps:

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter a name for the remote printer. This can be any name you choose.

```
Printer name:  papadoc ↵
```

3. Enter the type of connection, **Remote lpNet to lpd system**.

```
Connection type: [Local /dev/tty or /dev/lp] Remote
lpNet to lpd system ↵
```

4. Refuse the Quick Add feature.

```
Quick Add using default values? [yes] no ↵
```

5. Supply the printer type.

```
Printer type: [printer-80] PS-b
```

6. Enter the name of the remote host to which the printer is connected. (You must have added this host through Device -> Printer -> Systems -> Add as explained in the previous section, "Allowing host access to remote printer resources.")

```
Remote Host: netware_serv ↵
```

7. Enter the name of the remote printer as defined on the NetWare machine.

```
Remote Printer Name: psdoc ↵
```

8. Accept the default values for the following printer attributes:

```
Users to allow: [all] ↵
Users to deny: ↵
Forms to allow: [none] ↵
Forms to deny: ↵
Enable? [yes] ↵
Accept? [yes] ↵
```

9. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

```
Printer Description: Draft laser printer in lab 5.
```

The note accompanies the printer name when you list the status of the printer.

10. Confirm your entries. A warning appears from `sysadm` reminding you that your system must be added to the remote system and the remote printer must exist on the remote system.

```
OK to perform operation? [yes] ↵
You will not be able to print successfully unless
the system administrator for the remote system has
added your system via Device->Printer->Systems->Add
operation from sysadm. The remote printer will also
need to exist currently on the remote system.
papadoc has been added.
Printer papadoc has been enabled.
Printer papadoc has been set to accept requests.
```

The printer **papadoc** located on the NetWare system is now ready for remote printing.

Adding printers connected to SunOS systems

SunOS™ systems use the BSD method of printing, which uses the Line Printer Daemon (LPD). The DG/UX system supports remote printing to printers on these systems.

Adding printers connected to SunOS systems requires you to perform some tasks on both the SunOS and the DG/UX systems. This section describes both tasks.

Configuring SunOS systems for remote printing

Before performing the tasks in this section, be sure that the SunOS system is running LPD. You can check for the LPD daemon with the following command:

```
% ps -aux | grep lpd
```

If the LPD daemon is running, output from the above command will be similar to the following;

```
root 3467 0.0 52 0 ? IW 04:22 0:01 /usr/lib/lpd
```

If the daemon is not running, you can start it by logging in as root and issuing the following command:

```
# /usr/lib/lpd
```

When you are sure that the LPD daemon is running, you can configure the SunOS system for remote printing. You must perform the following tasks:

1. Log in to the SunOS machine as root and add the name and IP address of the DG/UX system to the **/etc/hosts** file.
2. Verify the local SunOS printer is working. If not, edit the **/etc/printcap** file, adding an entry for the local printer that is accepting print requests from the remote DG/UX system. For a line printer by the name of **doclp**, the entry could be as follows:

```
doclp|Line printer in lab 5:\  
:lp=/dev/lp:sd=/usr/spool/babydoc:
```

3. If the spool directory does not exist, make it as follows:

```
mkdir /usr/spool/babydoc
```

4. With a text editor, edit the **/etc/hosts.lpd** file, adding the name of the DG/UX system.

```
shaman
```

Configuring the DG/UX system for printing to SunOS systems

When you have completed the procedures on the SunOS system, you must do the following two tasks on the DG/UX system:

- Allow host access to the SunOS machine.
 - Add a printer on the SunOS machine to your client system.
- ▶ Allow host access to the SunOS machine

Allowing host access means that your local computer can use printers on the SunOS machine. To allow host access, you must register the SunOS machine on your local system.

1. Follow this path through `sysadm`:

Device -> Printer -> Systems -> Add

Sysadm guides you through a series of prompts. At each prompt, press `Enter` (`\n`) to accept the displayed default, or enter a new value.

2. Specify the names of systems to share printer services.

System name: **sun_set** `\n`

3. Choose the printer scheduler type used by the remote system. The scheduler type indicates the network protocol the remote system uses. Valid choices are **s5** for UNIX System V (the default) or **bsd** for BSD Line Printer Daemon (LPD) Protocol, SunOS, and Novell Netware.

Scheduler Type: [s5] **bsd** `\n`

4. Accept the default value for the number of minutes that the connection with the remote system may remain idle before terminating.

Connection Timeout: [10] `\n`

5. For better response, enter **2** as the value for the number of minutes to wait following a disconnection of service with a remote system before attempting to re-establish the connection.

Retry Period: [10] **2** `\n`

6. Enter a descriptive note about the remote system, such as the name of its system administrator, a phone number, or location.

Comment: **Sun system in development hallway.** `\n`

Do not use single quotation marks in the note. The note appears when you list information about the system using the `list` operation.

7. Confirm your entries:

OK to perform operation? [yes] `\n`

"sun_set" has been added.

After completing the procedures for allowing host access, you can add remote printers on your local system. To add a remote printer, gather remote host and printer information to provide **sysadm** and use **sysadm** to add the printer.

► Gather and record the following information. Use the forms in Appendix C.

remote host name Name of the remote SunOS system.

remote printer name Name of the printer on the remote SunOS system.

local printer name	Name you give the remote printer on your local machine. A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. To communicate with a SunOS system through lpNet, the connection type is Remote lpNet to lpd system .
printer type	The terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> . See Table 1-1 for a list of valid printer types.

Adding a remote line printer connected to a SunOS system

The following example uses the quick add feature to add a line printer with this configuration:

```
remote host:          sun_set
remote printer name:  doclp
local printer name    babydoc
connection type       Remote lpNet to lpd system
```

Follow these steps:

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter a name for the remote printer. This can be any name you choose.

```
Printer name:  babydoc ↵
```

3. Enter the connection type, **Remote lpNet to lpd system**.

```
Connection type: [Local /dev/tty or /dev/lp] Remote lpNet to lpd system ↵
```

4. Accept the Quick Add feature.

Quick Add using default values? [yes] ↵

5. Enter the name of the remote host to which the printer is connected. (You must have added this host through Device -> Printer -> Systems -> Add as explained in the previous section, "Allowing host access to remote printer resources.")

Remote Host: **sun_set** ↵

6. Enter the name of the remote printer. This is the name of the printer as defined on the SunOS machine.

Remote Printer Name: **doclp** ↵

7. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Line printer in lab 5.**

The note accompanies the printer name when you list the status of the printer.

8. Confirm your entries. A warning appears from sysadm reminding you that your system must be added to the remote system and the remote printer must exist on the remote system.

OK to perform operation? [yes] ↵

You will not be able to print successfully unless the system administrator for the remote system has added your system via Device->Printer->Systems->Add operation from sysadm. The remote printer will also need to exist currently on the remote system.
babydoc has been added.

Printer babydoc has been enabled.

Printer babydoc has been set to accept requests.

The printer **babydoc** located on the SunOS system is now ready for remote printing.

Adding remote printers connected to DG/UX systems and LPD systems on a Trusted DG/UX system

IMPORTANT: Data General offers B1 and C2 Trusted DG/UX systems. Both are optional and must be purchased in addition to the DG/UX system.

This section applies only if you have purchased one of the Trusted DG/UX systems.

This section describes how to add remote printers connected to DG/UX System 5.4 Release 2.01 (and later) on a Trusted DG/UX system. Some of the procedures here also apply to BSD systems.

When adding remote printers on a Trusted DG/UX system, the procedure and concepts are the same as presented in the previous sections of this chapter. For B1 systems, however, you must answer two additional prompts. The prompts are MAC Range Lo: and MAC Range Hi:. MAC Range Lo: indicates the lowest MAC label at which the printer can accept jobs. MAC Range Hi: is the highest MAC label at which the printer can accept jobs. For detailed information on MAC labels, see *Managing Security on the Trusted DG/UX™ System*.

To add a remote PostScript printer using lpNet, for example, follow the same procedures documented earlier in this chapter in “Adding remote printers connected to DG/UX systems and LPD systems.” That is, you do the following:

1. allow host access to remote printer resources,
2. gather printer information, and
3. add the printer using **sysadm**.

When you are ready to add the printer using **sysadm**, follow the procedures in the next section.

WARNING: If you overloaded this release on a previous security release, you must add your printer definitions again. If you overloaded this release on a regular DG/UX system, you must modify the print queues to include a MAC range. Do this by performing the modify operation in the the Device -> Printer -> Devices menu of **sysadm**. You will be prompted for the low and high values of the MAC range. Users cannot print to those queues until you modify them.

WARNING: If a system had remote printers added to the system before the Trusted DG/UX system was installed, these printers should be deleted before any lp-related activities are undertaken.

Adding a remote PostScript printer using lpNet

The following example uses this configuration:

Remote Host: babydoc

Remote Printer name: doc1
local printer name: psdoc
connection type: Remote lpNet to DG/UX 5.4. system
printer type: PS-b

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default or enter a new value.

2. Enter a name for the remote printer. This can be any name you choose.

Printer name: **psdoc** ↵

3. Supply the connection type. Enter **Remote lpNet to DG/UX 5.4 system** to indicate a DG/UX 5.4 system.

Connection type: [Local /dev/tty or /dev/lp] **Remote lpNet to DG/UX 5.4 system** ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type.

Printer type: [printer-80] **PS-b**

If you do not know the type of printer on the remote system, accept the default value (printer-80).

6. Enter the name of the remote host to which the printer is connected. (You must have added this host through Device -> Printer -> Systems -> Add as explained in the previous section, "Allowing host access to remote printer resources.")

Remote Host: **babydoc** ↵

7. Enter the name of the remote printer. This is the name of the printer as defined on the remote host.

Remote Printer Name: **doc1** ↵

8. Accept the default values for the following printer attributes:

Users to allow: [all] ↵
Users to deny: ↵
Forms to allow: [none] ↵
Forms to deny: ↵
Enable? [yes] ↵
Accept? [yes] ↵

9. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Production laser printer on doc hallway.**

The note accompanies the printer name when you list the status of the printer.

10. Supply the lowest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Lo: [IMPL_LO] ↵

11. Supply the highest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Hi: [ACR_HI] ↵

12. Confirm your entries. A warning appears from **sysadm** reminding you that your system must be added to the remote system and the remote printer must exist on the remote system.

OK to perform operation? [yes] ↵

You will not be able to print successfully unless the system administrator for the remote system has added your system via Device->Printer->Systems->Add operation from sysadm. The remote printer will also need to exist currently on the remote system.

psdoc has been added.

Printer psdoc has been enabled.

Printer psdoc has been set to accept requests.

End of Chapter

5

Configuring remote printers using the remshlp program

This chapter shows how to configure printers that are connected to DG/UX systems prior to Release 5.4 and to AOS/VS II systems. For both systems, the key is to set up the remote shell first, then add the remote printer with **sysadm**.

When adding remote printers, make sure that the printer is physically attached to the remote system you specify. The DG/UX system does not allow you to send print requests through one remote system to reach a printer attached to yet another remote system.

The DG/UX system does not allow you to define multiple remote queues for the same remote device. For example, let's suppose that you want use a printer, **lp1**, that is attached to a remote system called **hawk**. On your local system, **trax**, you can set up a queue (for example, **hawk|lp1**) to send files to the remote machine. Once you have established the queue **hawk|lp1**, you can not define another queue on **trax** to print to **lp1** on the remote system **hawk**.

Each of the two major sections in this chapter contains examples of how to configure printers. The example in "Adding printers attached to DG/UX systems prior to Release 5.4" shows how to configure a remote HP LaserJet printer. The example in "Adding remote printers attached to AOS/VS II systems" shows how to configure a remote PostScript printer.

IMPORTANT: Be sure to complete hardware installation before performing the procedures in the following sections.

Adding printers attached to DG/UX systems prior to Release 5.4

This section describes how to add remote printers that are attached to systems prior to DG/UX system 5.4. Adding printers to these systems requires that you:

- Enable remote printer access on DG/UX systems prior to Release 5.4.
- Add a Remote Printer Attached to a DG/UX system prior to Release 5.4.

Enabling remote printer access on DG/UX systems prior to 5.4

Before accessing remote printers on DG/UX systems prior to Release 5.4, you must enable remote printer access by providing

login permission on the remote system. You can do this by editing one of the files below or by using **sysadm**.

\$home/.rhosts The user's **.rhosts** file (located in the user's home directory) lists the names of the remote hosts the user can log in to. Logins to remote hosts are restricted to particular users. An entry in the user's **.rhosts** file assures a higher level of security than the **/etc/hosts.equiv** file.

/etc/hosts.equiv The host's **/etc/hosts.equiv** file lists the names of remote hosts the local host can connect to. The remote system does not prompt the user for a password when he issues **rlogin** and **remsh** commands. Logins to remote hosts are restricted to particular hosts. An entry in a host's **/etc/hosts.equiv** file provides a lower level of security than the user's **.rhosts** file.

To access printers on a remote DG/UX system prior to Release 5.4, make sure you have a home directory on the remote system. From your home directory on the remote system, create a **.rhosts** file if one does not exist. Edit this file to include the name of your local host. The file contains a list of names of remote hosts, with one host name per line starting in the first column. An example follows:

```
# more .rhosts ↵
gyramax
goose
mad-max
matilda
```

The hosts in this file are allowed to access this system.

To allow host access, create the **/etc/hosts.equiv** file, if it does not exist. Edit this file to include your local host and users to access printers. This file is also known as the trusted hosts database. A user is considered local if the name of his host appears in this file. A line that contains a hostname indicates that anyone logging in from that host is trusted. As the example below shows, you can edit **/etc/hosts.equiv** through **sysadm**.

To add the user **johnson** from the remote system **max-server** as a trusted user on a DG/UX system prior to Release 5.4, do the following:

1. Follow this path through **sysadm** and respond to the prompts:

```
Networking -> TCP/IP -> Databases -> Trusted Hosts
-> Add
```

2. Supply the name of the host to allow remote access.

Host Name: **max-server** ↵

3. Specify the authorized user who has login privileges.

User Name: [all] **johnson** ↵

4. Press Enter to accept the values that you have entered.

OK to perform operation? [yes] ↵
johnson on max-server has been added.

The user **johnson** on remote system **max-server** can access the host using the remote shell (**rsh** or **remsh**) or the remote login (**rlogin**). Refer to the **rsh(1)**, **remsh(1)**, and **rlogin(1)** manual pages for more information.

Adding a remote printer attached to a DG/UX system prior to 5.4 to your local system

After establishing remote printer access on the remote system, you can add remote printers on your local system. This section presents an example to show how to add a remote HP LaserJet printer. To add a remote printer, you must gather remote host and printer information and add the printer using **sysadm**.

- Gather and record the following information. Use the forms in Appendix C.

remote host name	Name of the remote system.
remote printer name	Name of the printer on the remote system.
local printer name	Name you give the remote printer on the local machine running the DG/UX system. A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.

connection type Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. To communicate with a system through **remsh**, the connection type is **Remote to system other than DG/UX 5.4**.

printer type Specifies the **terminfo** database file, which describes your printer's capabilities. The **terminfo** database files are located at **/usr/lib/terminfo/x/y**, where *x* is a single-letter name for the directory containing the printer type file *y*.

For example, the **terminfo** file **/usr/lib/terminfo/e/epsonfx** is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. Table 1-1 lists valid printer types.

Adding a remote HP LaserJet printer

This example adds a printer using a serial connection and HP LaserJet Series II emulation mode. It uses the full add feature and this configuration:

```
remote host:           termite
remote printer name:  hp3
local printer name    psdoc
connection type       Remote to system other than DG/UX 5.4.
printer type          hplaserjetII
```

Follow these steps:

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default or enter a new value.

2. Enter the local name you have chosen for the printer.

```
Printer name:  psdoc ↵
```

3. Supply the connection type. The connection type indicates the type of system to which the remote printer is attached. In this case, both BSD systems and DG/UX systems prior to 5.4 are indicated by the same connection type. Enter **Remote to system other than DG/UX 5.4** at the prompt.

Connection type: [Local /dev/tty or /dev/lp] **Remote to system other than DG/UX 5.4** ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type. For a list of available printer types, type the help character (?).

Printer type: [printer-80] **hplaserjetII** ↵

6. Supply the input type.

Input types: [simple] **laserjet** ↵

7. The next prompt asks for the interface script. **Sysadm** offers three remote model interface scripts:

remshlp Prints using the **lp** command.

remshlp_a Prints to AOS/VS II machines.

remshlp_bsd Prints using the **lpr** command.

Interface script: [remshlp] **remshlp_bsd** ↵

8. Supply the name of the remote host to which the printer is attached.

IMPORTANT: You should have already enabled remote access to this host through the **/.rhosts** or **/etc/hosts.equiv** file.

Remote Host: **termite** ↵

9. Enter the name of the remote printer.

Remote Printer Name: **hp3** ↵

10. Accept the defaults for the remaining prompts:

Stty options: [none] ↵

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Laser printer in Lab 2.**

The note accompanies the printer name whenever you list the status of added printers.

12. Confirm your choices.

```
OK to perform operation? [yes] ↵
You will not be able to print successfully unless
the system administrator for the remote system has
added your system via Networking->TCP/IP->Trusted
Hosts->Add operation from sysadm. The remote printer
will also need to exist currently on the remote
system.
psdoc has been added.
Printer psdoc has been enabled.
Printer psdoc has been set to accept requests.
```

Default remote remsh printer attributes

When adding remote printers with **remsh**, **sysadm** supplies default values for the less common printer attributes (Table 5-1).

Table 5-1 Default attributes for remote remsh printers

Printer attribute	Default value
Stty options	none
Print options	nobanner
Fault recovery	continue
Alert messages	mail
Alert interval	0
Forms to allow	none
Forms to deny	N/A
Users to allow	all
User to deny	N/A
Enable printer	yes
Accept printer	yes
Device	N/A
Device is a login	N/A

Adding remote printers attached to AOS/VS II systems

This section describes how to add remote printers that are attached to AOS/VS II systems. For printing through **remsh**, AOS/VS II systems accept print requests from the **lp** and **rlp** commands.

To use the **lp** command, the system administrator on the remote machine running the DG/UX system (or another UNIX system) must define a remote printer in **sysadm** and must have **lpsched** running on the DG/UX system. The **lpsched** program determines that the destination is a remote printer and uses **remsh** to send an **lp** command to the AOS/VS II system.

The **rlp** command, unlike the **lp** command, does not require that the remote printer be defined on the DG/UX system. It uses a shell script that invokes **remsh** to send the **lp** command to the AOS/VS II system.

The shell script is shipped with AOS/VS II TCP/IP and is moved to **/usr/bin** on the DG/UX system. Users on the DG/UX system invoke the **rlp** script with the print queue name to use on the AOS/VS II system. An **rlp** script also exists on the ECLIPSE® computer for printing to remote systems.

The next two subsections explain how to set up your DG/UX and AOS/VS II systems to use LP.

Setting up AOS/VS II systems for remote printing

On the AOS/VS II system, follow these steps:

1. If you plan to print **nroff** output files or other printer-formatted files on the AOS/VS II printer, add the following command sequence to the **:UP.CLI** macro.

```
Su! CX PAUSE @LPB ↵  
Su! CX BINARY @LPB :UTIL:FORMS:LPT.DEF.FORM ↵  
Su! CX CONTINUE @LPB ↵  
Su!
```

2. Using the **PREDITOR** utility, create two user accounts with the names of **LP** and **DAEMON** and do the following:
 - a. Use any encrypted password.
 - b. Limit the account to one child.
 - c. Assign Superuser access to the account.

- d. For better security, do not assign it Console, Batch, and Virtual Console accesses.
 - e. Assign Access Local Resources From Remote Machines access only if you also plan to set up your AOS/VS II system as a tape server.
 - f. Exit PREDITOR.
3. Delete the :UDD:LP directory, create a link to the :UDD:DAEMON directory, and give the LP and DAEMON accounts access to the files in :UDD:DAEMON directory. For example, type

```
Su! DIRECTORY :UDD ↵
Su! DELETE LP ↵
Su! CREATE/LINK LP :UDD:DAEMON ↵
Su! ACL :UDD:DAEMON:# DAEMON,OWARE,LP,OWARE ↵
Su!
```

4. Define each remote system that is to access the AOS/VS II printer.
 - Create the :UDD:DAEMON:.RHOSTS file and add the DG/UX hostnames and user names to it. Use the format

unix-hostname [username]

where

unix-hostame is the case-sensitive primary host name of a DG/UX system. Make an entry for each system.

The :ETC:HOSTS file lists the primary host name as the first name opposite the Internet address. Where multiple LAN boards exit, be sure you are addressing the correct board. The PING command also displays the primary host name.

username is a case-sensitive name of a DG/UX account. Make an entry for **lp**, **daemon**, **sysadm**, **root** and each DG/UX user account.

- Give DAEMON and LP owner access to the :UDD:DAEMON:.RHOSTS file. Use the following command:

```
Su! ACL :UDD:DAEMON:.RHOSTS DAEMON, OWARE
LP,OWARE ↵
Su!
```

If each DG/UX system user has an AOS/VS II account, do the following:

- Instruct each DG/UX system user to create the **.RHOSTS** file in his log-in directory or you can create an entry in the **:ETC:HOSTS.EQUIV** file for each user. Make an entry using the format

unix-hostname [username]

where

unix-hostame is the case-sensitive primary host name of a DG/UX system. Make an entry for each system.

The **:ETC:HOSTS** file lists the primary host name as the first name opposite the Internet address. Where multiple LAN boards exist, be sure you are addressing the correct board. The PING command also displays the primary host name.

username is a case-sensitive name of the user's remote account. A username placed after the hostname limits access to that user on that host.

Using the **:ETC:HOSTS.EQUIV** file can inadvertently give unauthorized access to your system. Using the **.RHOSTS** file in users' log-in directories limits the remote access.

IMPORTANT: If you list users in the **:ETC:HOSTS.EQUIV** file, privileged users must still create the **.RHOSTS** file in their log-in directory.

- Have each user set the ACL of his **:UDD:[!USERNAME]:.RHOSTS** file to **[!USERNAME], OWARE**. Use the command

) **ACL :UDD:[USER]:.RHOSTS [!USER], OWARE ↵**

5. Verify that the **:NET:UTIL:RSH.SEARCHLIST.CLI** macro contains the **:UTIL** and **:NET:UTIL** directories. If the file does not exist, copy the **:NET:UTIL:RSH.SEARCHLIST.CLI.PROTO** file to **:NET:UTIL:RSH.SEARCHLIST.CLI**.
6. Set the ACL of the **:NET:UTIL:RSH.SEARCHLIST.CLI** file to **+,RE**.

Adding printers from a remote AOS/VS II system

When you have completed the procedures on the AOS/VS II system, you can add the remote printer to the DG/UX system. To add the printer, you must do the following on the AViiON computer:

1. Gather host and printer information.
2. Test the remote shell.
3. Add the remote printer using **sysadm**.

- ▶ Gather and record the following information. Use the the forms in Appendix C.

remote host name	Name of the AOS/VS II system.
remote printer name	Name of the printer on the AOS/VS II system.
local printer name	Name you give the remote printer on the local DG/UX system. A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication (for example, local or networked) and determines the queries when adding or modifying a printer. The connection type for printers connected to AOS/VS II machines is Remote to system other than DG/UX 5.4 .
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> .

For example, the **terminfo** file **/usr/lib/terminfo/e/epsonfx** is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. Table 1-1 lists valid printer types.

Adding a remote PostScript printer

The following example uses this configuration:

```
remote host name:  miles
remote printer name:  post
local printer name:  remdoc1
Connection type:  Remote to system other than DG/UX 5.4
printer type:  PS-b
```

Testing the remote shell

Test the remote shell by logging in to the DG/UX system, using your username, and running the **remsh** command. If you do not have a

matching username and account on the remote AOS/VS II system, enter the following command:

remsh miles -l lp -n who

If you have a matching account on the AOS/VS II system, enter the following command:

remsh miles who

If there are no errors, you will receive a response from the AOS/VS II system similar to the following:

```
PID: 266 LP 000266 :CLI32.PR
```

You could, however, get one of the errors listed below. The list includes recommended solutions.

1. Permission Denied
Check the ACLs on **:UDD:DAEMON:+** and the hostnames in the **:UDD:DAEMON:.RHOSTS** file.
2. Try Again
Restart the **RDAEMON**.
3. rcmd: Socket: not owner
The setuid bit is not set on **/usr/bin/remsh**. As **root**, enter the following command: **chmod 04775 /usr/bin/remsh**.
4. Invalid Login
Check that the user is defined in the **.RHOSTS** file in the user's home directory if the user's usernames match or in **:UDD:DAEMON** if the user does not have a matching username.

Adding the remote printer

As root, use **sysadm** to create a remote printer. This associates a local logical printer with an AOS/VS II queue on the remote machine. Follow these steps:

1. In **sysadm**, follow this path:

```
Device -> Printer -> Devices -> Add
```

2. Enter the printer name. You can choose the name for this printer.

```
Printer name: remdoc1 ↵
```

3. Enter the connection type. Enter **Remote to system other than DG/UX 5.4**.

```
Connection type: [Local /dev/tty or /dev/lp] Remote to system other than DG/UX 5.4 ↵
```

4. Accept the Quick Add feature.

Quick Add using default values? [yes] ↵

5. Supply the printer type. Enter the help character (?) for a list of available printer types.

Printer type: [printer-80] **PS-b** ↵

6. Supply the input type. Enter the help character (?) for a list of input types.

Input types: [simple] **PS** ↵

7. Enter the name of the interface script. Enter the help character (?) for a list of interface scripts.

Interface script: [remshlp] **remshlp_a** ↵

8. Supply the name of the remote host to which the printer is attached.

IMPORTANT: You should have already enabled remote access to this host through the **.rhosts** or **/etc/hosts.equiv** file.

Remote Host: **miles** ↵

9. Enter the name of the remote printer.

Remote Printer Name: **post** ↵

10. Accept the default values for the following printer attributes:

Stty options: [none] ↵
Print options: [nobanner] ↵
Stty options: [-clocal] ↵
Fault recovery: [continue] ↵
Alert messages: [mail] ↵
Alert interval: [0] ↵
Users to allow: [all] ↵
Users to deny: ↵
Forms to allow: [none] ↵
Forms to deny: ↵
Enable? [yes] ↵
Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Laser printer on on doc hallway.**

The note accompanies the printer name whenever you list the status of added printers.

12. Confirm your choices.

```
OK to perform operation? [yes] ↓
You will not be able to print successfully unless
the system administrator for the remote system has
added your system via Networking->TCP/IP->Trusted
Hosts->Add operation from sysadm. The remote printer
will also need to exist currently on the remote
system.
remdoc1 has been added.
Printer remdoc1 has been enabled.
Printer remdoc1 has been set to accept requests.
```

13. If the queue on the ECLIPSE computer is a PLOT queue rather than a PRINT queue, edit the script file in **/etc/lp/interfaces** that matches the name of the printer. Search for this line:

```
/usr/bin/remsh ${rem_host} -l lp lp -d${rem_printer}
${title} -n${copies} -oDEST=${user_name} ${options}
2>&l'
```

Delete the following:

```
-oDEST=$ {user_name}
```

14. Log in to the AOS/VS II system and test the printer by issuing the following command:

```
lp -d remdoc1 testfile
```

15. Check the status of the printer.

```
lpstat -t
```

If the following error appears, check the **:UDD:DAEMON:RSH.SETUP.CLI** to make sure the searchlist is set correctly and the ACL (access control list) is correct.

```
Not a command, macro, or program
```

16. After correcting any errors, enter **sysadm** to re-enable the printer.

Default attributes for remote remsh printers on AOS/VS II systems

When using **remsh** to add remote printers attached to computers running the AOS/VS II system, **sysadm** supplies default values for the less common printer attributes (Table 5-2).

Table 5-2 Default attributes for remote remsh printers on AOS/VS II

Printer attribute	Default value
Stty options	none
Print options	nobanner
Fault recovery	continue
Alert messages	mail
Alert interval	0
Forms to allow	none
Forms to deny	N/A
Users to allow	all
User to deny	N/A
Enable printer	yes
Accept printer	yes
Device	N/A
Device is a login	N/A

Adding printers attached to DG/UX systems prior to Release 5.4 on a Trusted DG/UX system

IMPORTANT: Data General offers B1 and C2 Trusted DG/UX systems. Both are optional and must be purchased in addition to the DG/UX system. This section applies only if you have purchased one of the Trusted DG/UX systems.

This section describes how to add remote printers that are attached to systems prior to DG/UX system 5.4 on a trusted system. When adding printers to these systems, the concepts and procedures are the same as those documented earlier in “Adding printers attached to DG/UX systems prior to Release 5.4.” That is, you must:

- Enable remote printer access on DG/UX systems prior to Release 5.4.
- Add a Remote Printer Attached to a DG/UX system prior to Release 5.4.

On B1 trusted systems, however, you must answer two additional prompts. The prompts are MAC Range Lo: and MAC Range Hi:.. MAC Range Lo: indicates the lowest MAC label at which the printer can accept jobs. MAC Range Hi: is the highest MAC label

at which the printer can accept jobs. For detailed information on MAC labels, see *Managing Security on the DG/UX™ System*.

WARNING: If you overloaded this release on a previous security release, you must add your printer definitions again. If you overloaded this release on a regular DG/UX system, you must modify the print queues to include a MAC range. Do this by performing the modify operation in the the Device -> Printer -> Devices menu of sysadm. You will be prompted for the low and high values of the MAC range. Users cannot print to those queues until you modify them.

WARNING: If a system had remote printers added to the system before the Trusted DG/UX system was installed, these printers should be deleted before any lp-related activities are undertaken.

Adding a remote HP LaserJet printer on a Trusted system

This example adds a printer using a serial connection and HP LaserJet Series II emulation mode. It uses the full add feature and this configuration:

```
remote host:          papadoc
remote printer name: hp3
local printer name:   psdoc
connection type:      Remote to system other than DG/UX 5.4.
printer type:         hplaserjetII
```

Follow these steps:

1. Follow this path through sysadm:

```
Device -> Printer -> Devices -> Add
```

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default or enter a new value.

2. Enter the local name you have chosen for the printer.

```
Printer name:  psdoc ↵
```

3. Supply the connection type. The connection type indicates the type of system to which the remote printer is attached. In this case, both BSD systems and DG/UX systems prior to 5.4 are indicated by the same connection type. Enter Remote to system other than DG/UX 5.4 at the prompt.

Connection type: [Local /dev/tty or /dev/lp] **Remote to system other than DG/UX 5.4** ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type. For a list of available printer types, type the help character (?).

Printer type: [printer-80] **hplaserjetII** ↵

6. Supply the input type.

Input types: [simple] **laserjet** ↵

7. The next prompt asks for the interface script. **Sysadm** offers three remote model interface scripts:

remshlp Prints using the **lp** command.

remshlp_a Prints to AOS/VS II machines.

remshlp_bsd Prints using the **lpr** command.

Interface script: [remshlp] **remshlp_bsd** ↵

8. Supply the name of the remote host to which the printer is attached.

IMPORTANT: You should have already enabled remote access to this host through the **./rhosts** or **/etc/hosts.equiv** file.

Remote Host: **papadoc** ↵

9. Enter the name of the remote printer.

Remote Printer Name: **hp3** ↵

10. Accept the defaults for the remaining prompts:

Stty options: [none] ↵

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

11. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Laser printer in Lab 2.**

The note accompanies the printer name whenever you list the status of added printers.

12. Supply the lowest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Lo: [IMPL_LO] ↵

13. Supply the highest MAC label at which the printer can accept jobs. You can accept the default.

MAC Range Hi: [ACR_HI] ↵

14. Confirm your choices.

OK to perform operation? [yes] ↵

You will not be able to print successfully unless the system administrator for the remote system has added your system via Networking->TCP/IP->Trusted Hosts->Add operation from sysadm. The remote printer will also need to exist currently on the remote system.

psdoc has been added.

Printer psdoc has been enabled.

Printer psdoc has been set to accept requests.

End of Chapter

6

Configuring remote printers connected to the network

This chapter shows how to configure printers that are connected to the network by term servers and print servers. The DG/UX system supports the following network devices:

- CS/2000 Series TermServer
- CS/2100 Series TermServer
- CS/3000 Series TermServer
- CS/3100 Series TermServer
- HP LaserJet 4 printer with a network adapter card
- MicroAnnex XL Host-boot Termserver
- MicroAnnex XL Self-boot Termserver
- Annex Three Host-boot Termserver
- Annex Three Self-boot Termserver
- Datability VCP300/1000 TCP Series Termserver
- MiLAN 3100 FastPort print servers

The DG/UX system provides the following interface scripts for the networked devices above:

termprinter_3com3.3	supports CS/2000 and 2100 Series TermServers.
termprinter_tcp	supports CS/3000 and 3100 Series TermServers and HP LaserJet 4 printers with network adapter cards.
termprinter_telnet	supports printers connected to a specific telnet port.
dg_annex	supports both the Annex and MicroAnnex Term servers. This is the default.
datability	supports the Datability VCP300/1000 TCP Series Termserver.
fastport	supports the MiLAN 3100 FastPort print servers.

This chapter contains the following examples:

- Adding a plotter connected to a CS/2000 Series TermServer.

- Adding a PostScript printer connected to an Annex termserver.
- Adding an HP LaserJet 4 printer with HP JetDirect
- Adding a PostScript printer connected to a FastPort print server.
- Adding a PostScript printer connected to a FastPort print server on a Trusted system

When configuring a networked printer to accept documents that include 8-bit data, you will have to modify some of the defaults. For example, you must set the **stty** option **-opost**. This option turns off output processing.

IMPORTANT: Be sure to complete hardware installation before performing the procedures in the following sections.

Printers connected to termserver

Termserver support serial connections. When connecting printers to termserver, you have the option to set low-level communications, such as flow control, baud rate, and parity. The interface script is set to use software flow control. If, for example, you have chosen to use hardware flow control, remember to turn off software flow control and turn on hardware flow control, using the **stty** options. You can make these changes in **sysadm**.

Adding printers to termserver

To add a printer to a termserver, you need to gather printer and termserver information for **sysadm**. Then, you need to define the termserver for the DG/UX system and add the printer using the **sysadm** utility.

- ▶ Gather and record the following information. Use the forms in Appendix C.

printer name	A printer name can contain these characters: a-z A-Z 0-9 _ . The name should begin with a letter or digit and be no more than 14 characters long.
printer address	The Internet address for the printer as assigned during the hardware setup.
connection type	Specifies the type of connection (for example, local or networked) and determines the queries when adding or modifying a printer. For termserver, the connection type is Remote to Networked Printer Device .

device name	The name of the termserver as it appears in the /etc/hosts file.
port number	The printer's port number on the termserver. See your termserver setup and documentation to determine the termprinter interface script port number. The telnet port 23 is the default port.
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> .

For example, the **terminfo** file **/usr/lib/terminfo/e/epsonfx** is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. Table 1-1 lists valid printer types.

Defining a termserver

Before you can begin using printers attached to a termserver, you must define the termserver for the DG/UX system. The following example shows the **sysadm** ASCII interface.

To define a termserver, you must place the termserver name and the Internet address associated with the physical port on the termserver in the **/etc/hosts** file or YP database if you are using NIS.

Follow these steps:

1. Follow this path through **sysadm**:

```
Networking -> TCP/IP -> Hosts -> Add
```

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the hostname.

```
Host: monk ↵
```

3. Enter the Internet address.

```
Internet address: 128.999.100.01 ↵
```

4. Enter any aliases you want.

Alias List: ↵

5. Confirm your choices.

Ok to perform operation [yes]:↵

Adding a plotter connected to a termserver

The following example adds a plotter that is connected to a Data General TermServer 2000. The **termprinter_3com3.3** interface script supports this termserver.

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

2. Enter the printer name.

Printer name: **net_plotter1** ↵

3. Supply the connection type.

Connection type: [Local /dev/tty or /dev/lp]
Remote to networked printer device ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Accept the default printer type or choose from the list presented in the help message.

Printer type: [printer-80] ↵

6. Accept the default input type.

Input types: [simple] ↵

The default, `simple`, refers to an ASCII file.

7. Accept the default interface script.

Interface script: [dg_annex] ↵

8. Enter the name of the printer device as it appears in the **/etc/hosts** file.

Networked printer device name: **monk**

9. Enter the printer's port number on the termserver. Use telnet port 23, which is the default termprinter interface script port, or consult your termserver setup and documentation to determine the port number.

Port number or name: **23**

10. Turn off output processing by entering the **stty** option **-opost**.

Stty options: **-opost**

11. Enter the **banner** option if you want each print job to print first a page identifying the user.

Print options: [nobanner] **banner** ↵

12. Accept the defaults for the following attributes:

Device is also a login terminal: [no] ↵

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

13. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **Big plotter in prototype lab.** ↵

14. Confirm your choices:

OK to perform operation? [yes] ↵

net_plotter1 has been added.

Printer net_plotter1 has been enabled.

Printer net_plotter1 has been set to accept requests.

Your plotter is ready for operation.

Adding a PostScript printer connected to an annex termserver

This example adds a printer that accepts PostScript files and uses hardware flow control. Before performing the procedures below, gather information for **sysadm** and define the termserver for the DG/UX system as demonstrated in the above example. Follow these steps:

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

Sysadm guides you through a series of prompts. At each prompt, press Enter (↵) to accept the displayed default, or enter a new value.

2. Enter the printer name.

Printer name: **laser_net2** ↵

3. Supply the connection type.

Connection type: [Local /dev/tty or /dev/lp]
Remote to networked printer device ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Enter the PostScript printer type. See the list presented in the help message.

Printer type: [printer-80] **PS** ↵

6. Enter **PS** (PostScript) for the input type. Input types represent the final formats that the printer can accept and affect how your print request is filtered.

Input types: [simple] **PS** ↵

7. Accept the default interface script .

Interface script: [dg_annex] ↵

8. Enter the name of the printer device as recorded in **/etc/hosts**. This entry is either the name of the termserver or the name of the port. This example uses the name of the termserver.

Networked printer device name: **termserver3**

9. Enter the printer's port number. The tcpip port number is 70xx, where xx represents the termserver's physical port. In this example, the physical port is 3, so the port number is 7003.

Port number or name: **7003**

10. You need to turn off output processing, turn off software flow control, and turn on CTS hardware flow control. Enter the **stty** options **-opost**, **-ixon**, **-ixoff**, and **ctsxon**.

Stty options: [clocal] **-opost -ixon -ixoff ctsxon**

11. Enter the banner option if you want each print job to print first a page identifying the user.

Print options: [nobanner] **banner** ↵

12. Accept the defaults for the following attributes:

Device is also a login terminal: [no] ↵

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

13. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **PostScript printer on annex termserver.** ↵

14. Finally, confirm your choices:

OK to perform operation? [yes] ↵

laser_net2 has been added.

Printer laser_net2 has been enabled.

Printer laser_net2 has been set to accept requests.

Your printer is ready for operation.

Adding HP LaserJet 4 printers with HP JetDirect

To add an HP LaserJet 4 printer with HP JetDirect, a network adapter card, gather printer and network information. Then assign an IP address and hostname to the printer on the printer's console and add the printer to the DG/UX system using **sysadm**.

- Gather and record the following information. Use the forms in Appendix C.

IP address	The Internet address for the print server. This is supplied by the network administrator. For information on Internet addresses, see <i>Managing TCP/IP on the DG/UX™ System</i> .
------------	--

host name	The name mapped to the Internet address assigned to the network card as it appears in the /etc/hosts file.
printer name	A printer name can contain these characters: a-z, A-Z, 0-9, _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication (for example, local or networked) and determines the queries the users sees when adding or modifying a printer. For the FastPort print server, the connection type is Remote to networked printer device .
port number	The printer's port number. which is 9100 for the HP LaserJet 4 printer.
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> . For example, the terminfo file /usr/lib/terminfo/e/epsonfx is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. Table 1-1 lists valid printer types.

The following example uses this configuration:

```
Host name:      hplaser4
IP address     128.999.44.444
```

Assigning an IP address to the printer

Follow these steps:

1. Follow this path through **sysadm**:

```
Networking -> TCP/IP -> Hosts -> Add
```

2. Add the printer to the local hosts database (**/etc/hosts**) or the Network Information Service (NIS) if your host is the NIS master. For information on the NIS, see *Managing ONC™/NFS® and Its Facilities on the DG/UX™ System*.

```
Host Name: hplaser4 ↵
```

3. Add the Internet address that your network administrator assigned.

Internet Address: **128.999.44.444** ↵

4. As an option, you can add an alias.

Alias list: ↵

5. Confirm your choices.

OK to perform operation? [yes] ↵
fastport2-pr has been added.

Adding an HP LaserJet 4 printer

To add the printer, use the **sysadm** program. We can use the quick add feature, accepting the default values. The printer uses the following values from above;

Host name: **hplaser4**
IP address **128.999.44.444**

Follow these steps:

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

2. Enter the printer name. For this example, enter **doc_ps1**.

Printer name: **doc_ps1**

3. Supply the connection type.

Connection type: [Local /dev/tty or /dev/lp]
Remote to networked printer device ↵

4. Accept the Quick Add feature.

Quick Add using default values? [yes] ↵

5. Supply the printer type. Use **PS-b** for a PostScript printer.

Printer type: [printer-80] **PS-b** ↵

6. Supply the input type. For PostScript printers, the only valid input type is **PS**.

Input types: [simple] **PS**

7. Enter the interface script. For the HP LaserJet 4 printer, the interface script is **termprinter_tcp**.

Interface script: [dg_annex] **termprinter_tcp** ↵

8. Enter the name of the printer device. For a list of devices attached to your system, check the file `/etc/hosts` or use `sysadm` to list the host names that NIS provides.

Networked printer device name: **hplaser4** ↵

9. Enter the port number of the printer. The port number of the HP LaserJet 4 printer is 9100.

Port number or name: **9100** ↵

10. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **PostScript production printer.** ↵

11. Confirm your choices:

OK to perform operation? [yes] ↵

doc_ps1 has been added.

Printer doc_ps1 has been enabled.

Printer doc_ps1 has been set to accept requests.

Default attributes for remote printers on network devices

When adding remote printers attached to network devices with the Quick Add feature, `sysadm` supplies default values for the less common printer attributes. Table 6-1 lists those default values.

Table 6-1 Default attributes for networked printers

Printer attribute	Default value
Stty options	none
Print options	nobanner
Device is a login	N/A
Fault recovery	continue
Alert messages	mail
Alert interval	0
Users to allow	all
Users to deny	N/A
Forms to allow	N/A
Forms to deny	N/A
Enable printer	yes
Accept printer	yes
Device	N/A

Printers connected to FastPort print servers

The FastPort network print server connects printers directly to TCP/IP Ethernet networks, using 10BaseT, 10Base2, or 10Base5 Ethernet media. You can reconfigure this device without taking it off the network either by connecting a terminal to the serial port or by making a telnet connection to its maintenance port. The FastPort has a number of on-board diagnostics and utilities that you can access in this manner. The FastPort has two printer ports, one parallel and one serial, and it can print through both ports simultaneously.

IMPORTANT: DG/UX System 5.4 Release 2.10 is the earliest revision of the DG/UX system that has built-in support for the FastPort.

When connecting parallel printers to the FastPort, the low-level communications, such as flow control, are defined by the Centronics interface. You don't have the option of changing these values. When connecting serial printers, however, you can change the low-level communications. If, for example, you have chosen to use hardware flow control, remember to turn off software flow control and turn on hardware flow control, using the `stty` options. You can make these changes in `sysadm`.

IMPORTANT: When using a DG 6480 PostScript printer, set the FastPort parallel printer port to **standard**. The default setting, **fast-DMA**, causes these printers to lose data.

Adding printers connected to FastPort print servers

To add a printer connected to a FastPort print server, gather printer, network, and print server information. Then assign an IP address to the print server and add the printer using **sysadm**.

- ▶ Gather and record the following information. Use the forms in Appendix C.

IP address	The Internet address for the print server. This is supplied by the network administrator. For information on Internet addresses, see <i>Managing TCP/IP on the DG/UX™ System</i> .
printer name	A printer name can contain these characters: a-z, A-Z, 0-9, _ . The name should begin with a letter or digit and be no more than 14 characters long.
connection type	Specifies the type of communication (for example, local or networked) and determines the queries the users sees when adding or modifying a printer. For the FastPort print server, the connection type is Remote to networked printer device .
host name	The name of the print server as it appears in the /etc/hosts file.
port number	The printer's port number on the print server.
printer type	Specifies the terminfo database file, which describes your printer's capabilities. The terminfo database files are located at /usr/lib/terminfo/x/y , where <i>x</i> is a single-letter name for the directory containing the printer type file <i>y</i> .

For example, the **terminfo** file **/usr/lib/terminfo/e/epsonfx** is provided for the printer type "epsonfx." The generic "printer-80" printer type can be used as a default for 80-column line printers. Table 1-1 lists valid printer types.

The following example uses this configuration:

Host name: fastport2-pr
IP address 128.999.55.555
Connection: parallel
Printer: DG Model 6773 (PostScript)

Assigning an IP address to the print server

Follow these steps:

1. Follow this path through sysadm:

Networking -> TCP/IP -> Hosts -> Add

2. Add the FastPort print server to the local hosts database (**/etc/hosts**) or the Network Information Service (NIS) if your host is the NIS master. For information on the NIS, see *Managing ONC™/NFS® and Its Facilities on the DG/UX™ System*.

Host Name: **fastport2-pr** ↵

3. Add the Internet address that your network administrator assigned.

Internet Address: **128.999.55.555** ↵

4. As an option, you can add an alias.

Alias list: ↵

5. Confirm your choices.

OK to perform operation? [yes] ↵
fastport2-pr has been added.

Adding a PostScript printer connected to a FastPort print server

This example adds a printer using a serial connection and hardware flow control. The **sysadm** program offers a quick add feature and full add feature. Because you need to set **stty** options, you must use the full add feature. The following example uses the sample configuration below:

Host name: fastport3-pr
IP address 128.999.55.555
Connection: serial
Printer: DG Model 6773 (PostScript)

Before performing the procedures below, be sure to assign an IP address to the print servers as demonstrated above.

Follow these steps:

1. Follow this path through **sysadm**:

Device -> Printer -> Devices -> Add

2. Enter the printer name. For this example, enter **doc_ps1**.

Printer name: **doc_ps1**

3. Supply the connection type.

Connection type: [Local /dev/tty or /dev/lp]
Remote to networked printer device ↵

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type. Use **PS-b** for a PostScript printer.

Printer type: [printer-80] **PS-b** ↵

6. Supply the input. For PostScript printers, the only valid input type is **PS**.

Input types: [simple] **PS**

7. Enter the interface script. For the FastPort print server, the interface script is **fastport**.

Interface script: [dg_annex] **fastport** ↵

8. Enter the name of the printer device. For a list of devices attached to your system, check the file **/etc/hosts** or use **sysadm** to list the host names that NIS provides.

Networked printer device name: **fastport2-pr** ↵

9. Enter the port number of the printer on the network print server. The port number of the FastPort print server is 2000 for the parallel printer and 2001 for the serial printer.

Port number or name: **2001** ↵

10. Because **stty** settings are ignored, accept the default.

Stty options: ↵

11. Enter the banner option if you want each print job to first print a page identifying the user.

Print options: [nobanner] **banner** ↵

12. Accept the defaults for the following attributes:

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

13. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **PostScript production printer on Fastport print server.** ↵

14. Confirm your choices:

OK to perform operation? [yes] ↵

doc_ps1 has been added.

Printer doc_ps1 has been enabled.

Printer doc_ps1 has been set to accept requests.

Remote networked printers on a Trusted DG/UX system

IMPORTANT: Data General offers B1 and C2 Trusted DG/UX systems. Both are optional and must be purchased in addition to the DG/UX system. This section applies only if you have purchased one of the Trusted DG/UX systems.

When adding networked printers on a Trusted DG/UX system, the procedure and concepts are the same as those documented in the previous sections of this chapter. For B1 systems, however, you must answer two additional prompts. The prompts are MAC Range Lo: and MAC Range Hi:. MAC Range Lo: indicates the lowest MAC label at which the printer can accept jobs. MAC Range Hi: is the highest MAC label at which the printer can accept jobs. For detailed information on MAC labels, see *Managing Security on the DG/UX™ System*.

To add a PostScript printer connected to a FastPort print server, for example, follow the same procedure presented above in “Adding printers connected to FastPort print servers.” That is,

1. gather printer, network, and print server information,
2. assign an IP address to the print server, and
3. add the printer using **sysadm**.

When you are ready to add the printer using **sysadm**, follow the procedures in the next section.

WARNING: If you overloaded this release on a previous security release, you must add your printer definitions again. If you overloaded this release on a regular DG/UX system, you must modify the print queues to include a MAC range. Do this by performing the modify operation in the the Device -> Printer -> Devices menu of **sysadm**. You will be prompted for the low and high values of the MAC range. Users cannot print to those queues until you modify them.

Adding a PostScript printer connected to a FastPort print server

This example adds a printer using a serial connection and hardware flow control. The **sysadm** program offers a quick add feature and full add feature. Because you need to set **stty** options, you must use the full add feature. The following example uses the sample configuration below:

```
Host name:      fastport3-pr
IP address:    128.999.55.555
Connection:    serial
Printer:       DG Model 6773 (PostScript)
```

Before performing the procedures below, be sure to assign an IP address to the print servers as shown earlier in "Assigning an IP address to the print server."

Follow these steps:

1. Follow this path through **sysadm**:

```
Device -> Printer -> Devices -> Add
```

2. Enter the printer name. For this example, enter **doc_ps1**.

```
Printer name: doc_ps1
```

3. Supply the connection type.

```
Connection type: [Local /dev/tty or /dev/lp]
Remote to networked printer device ↵
```

4. Refuse the Quick Add feature.

Quick Add using default values? [yes] **no** ↵

5. Supply the printer type. Use **PS-b** for a PostScript printer.

Printer type: [printer-80] **PS-b** ↵

6. Supply the input. For PostScript printers, the only valid input type is **PS**.

Input types: [simple] **PS**

7. Enter the interface script. For the FastPort print server, the interface script is **fastport**.

Interface script: [dg_annex] **fastport** ↵

8. Enter the name of the printer device. For a list of devices attached to your system, check the file **/etc/hosts** or use **sysadm** to list the host names that NIS provides.

Networked printer device name: **fastport2-pr** ↵

9. Enter the port number of the printer on the network print server. The port number of the FastPort print server is 2000 for the parallel printer and 2001 for the serial printer.

Port number or name: **2001** ↵

10. Because **stty** settings are ignored, accept the default.

Stty options: ↵

11. Enter the banner option if you want each print job to first print a page identifying the user.

Print options: [nobanner] **banner** ↵

12. Accept the defaults for the following attributes:

Fault recovery: [continue] ↵

Alert messages: [mail] ↵

Alert interval: [0] ↵

Users to allow: [all] ↵

Users to deny: ↵

Forms to allow: [none] ↵

Forms to deny: ↵

Enable? [yes] ↵

Accept? [yes] ↵

13. Supply a printer description. This might include the type of printer, the name of the system administrator, his or her phone number, or the location of the printer. This entry is optional; you can leave it blank.

Printer Description: **PostScript production printer on Fastport print server.** ↵

14. Supply the lowest MAC label at which the printer can accept jobs.
You can accept the default.

MAC Range Lo: [IMPL_LO] ↵

15. Supply the highest MAC label at which the printer can accept jobs.
You can accept the default.

MAC Range Hi: [ACR_HI] ↵

16. Confirm your choices:

OK to perform operation? [yes] ↵

doc_ps1 has been added.

Printer doc_ps1 has been enabled.

Printer doc_ps1 has been set to accept requests.

End of Chapter

7

sysadm management menus

This chapter explains the operations that the **sysadm** facility provides for setting up and managing printers. These operations appear in the Printer menu, which is located in the Device menu. The Printer menu operations are implemented using the commands and files discussed in Chapter 8.

For information on using the **lp** command, see the manual page **lp(1)**.

LP management procedures

This section discusses the operations that **sysadm** offers for managing the LP service. For information on using **sysadm**, invoke **sysadm** and see the Help menu.

To set your LP service up for the first time, start by adding devices. In the **sysadm** Printer menu, select the Devices menu. See “Managing Devices” for details. Before adding any remote printers, you must add entries for remote hosts using the Systems menu.

The Printer menu comprises the following selections:

- | | |
|---------|---|
| Devices | This menu contains operations to add, delete, modify, and list entries for the printers on your system. These entries determine not only what your printers are called and how the LP service should communicate with them, they also determine other useful features such as what kinds of jobs the printers can accept, how the LP service should handle a printer fault, and so on. You can also name a printer on your system to be the default printer. You can set the LP service to accept or reject requests submitted to a given printer or class, and you can enable and disable individual printers. |
| Classes | This menu contains operations to create, modify, remove, accept, reject, and list status of printer classes. |
| Filters | This menu contains operations to add, modify, delete, restore, and list filter programs for use with printers on your system. |

Forms	This menu contains operations to create, modify, remove, and list definitions for pre-printed forms used on your printers. You can also mount forms on printers, which means configure a printer to print that form, and unmount forms from printers. In addition, you can display the current mount status of defined forms.
Priorities	This menu contains operations to set the default priority for print requests. You can also set and remove priorities for individual users, and you can set a default priority for print requests. A list operation displays current priority settings.
Requests	This menu contains operations to list and cancel job requests as well as to hold requests and to resume held requests. There is also an operation for moving requests from one printer to another.
Scheduler/Service	This menu contains operations to start and stop the LP service scheduler and to display the current status of the scheduler.
Systems	For systems on a network, this menu contains operations to add, modify, remove, and list entries for remote systems with which you can share printer services.
List	This operation displays the status of the LP service, including LP service database information about the default printer setting, devices, classes, remote systems, forms, filters, and current status of the scheduler, devices, classes, forms, and jobs requests.

You need to be **root** or **lp** (uid 0 or 6) to use these operations, except for the Requests menu operations that cancel, hold, and resume print jobs. Except for the List operation in the Filters menu, you do not need to be superuser to invoke any of the List operations.

The following sections discuss the Printer menu operations and LP services concepts in more detail.

Managing printer devices

Before you can perform any printer-related operations, you need to define the printers that are available on the system. If you are adding printers connected to the local system, use the operations in

the Device -> Printer -> Devices menu. If you are adding printers connected to other systems in your network, first add the remote systems with the Device -> Printer -> Systems -> Add operation before proceeding to the Device -> Printer -> Devices menu.

This section discusses the various operations that **sysadm** offers for printer device management:

Add	This operation adds a printer device entry to the LP service databases.
Default	This operation allows you to define the default printer for requests that do not specify a destination.
Delete	Use this operation to remove the file, directories, and database entries that support a particular printer.
Modify	Use this operation to change the attributes of an existing printer.
Accept	This operation allows a printer to accept print requests.
Reject	This operation makes the LP service refuse requests for the printer. If a user submits a job for printing at this printer, the lp command returns an error.
Disable	This operation causes the LP service to cease sending jobs to a printer.
Enable	This operation tells the LP service to verify that a printing device is functional and if so to make it available for printing.
List	This operation produces a general status listing for your system's printing devices.

The following sections discuss these operations in more detail.

Adding printer devices

If you are adding a remote printer or printer class, you need to verify that you have already added an entry for the remote system using the following operation Device -> Printer -> Systems -> Add.

After you have added the system entry, use the Add operation documented here to set up access to the printer.

To declare a printer device to the LP service, use the Add operation in the Devices menu. This operation not only sets up the files and directories that the LP service needs to use the printer, it also lets you customize the printer's behavior.

You can add a printer to your system in two ways: Quick Add or Full Add. If you want to get a printer up and running immediately, you should choose Quick Add. Quick Add minimizes the number of prompts you have to answer by using established defaults for most of a printer's characteristics. If you choose Full Add, you must specify all of a printer's characteristics.

When you select Device -> Printer -> Systems -> Add, you must answer the following three queries:

Printer name A printer name may be any name that you choose. You should choose a name that tells something about the printer, such as what type of printer it is, its location at your workplace, or its location on your network. It is also a good idea to choose a name that is short and easy to remember.

Connection type This is the type of printer that you want to add. The connection types are:

Local /dev/tty or /dev/lp, which is a printer connected to your system through a device whose name begins with /dev.

Remote lpNet to DG/UX 5.4 system, which is a remote printer on a DG/UX Release 5.4 or later system running the lpNet daemon.

Remote lpNet to lpd system, which is a remote printer on a system that supports the Line Printer Daemon Protocol (lpd), also known as the Berkeley print spooler.

Remote to system other than DG/UX 5.4, which is a remote printer on a system that is not running DG/UX Release 5.4 or later. This type supports **remshlp**, which is a remote UNIX host that supports the **lp** command using **remsh**; **remshlp_bsd**, which is a remote UNIX host that supports the **lpr** command using **remsh**; and **remshlp_a**, which is a remote AOS/VS II host with TCP/IP II installed that supports the **lp** command using **remsh**.

Remote networked printer device, which is a remote printer connected to a network device independent of a host. This type supports the termprinter interface scripts (**termprinter_3com3.3**, **termprinter_tcp**, and **termprinter_telnet**), which are used for termserver

connections; the **fastport** script, which is used for MiLAN Fastport networked printer servers with a serial and parallel port; the **dg_annex** script, which is used for Annex and MicroAnnex term servers; and the **datability** script, which is used for datability term servers.

Quick Add using default values

Select this option if you want to use Quick Add to add a printer to your system. Quick Add is covered in more detail in the next section. If you do not select this option, you have to do a Full Add, which is covered in a later section.

Quick Add —

The Quick Add operation enables you to get a printer up and running immediately. Quick Add minimizes the number of queries you have to answer for each connection type by taking some standard defaults for the majority of a printer's characteristics. Each connection type has its own set of queries for adding a printer. See Chapters 3–6 for additional examples of Quick Add.

The following sections cover the Quick Add queries and the defaults associated with each connection type. For a complete description of each printer characteristic, see "Full Add."

Local /dev/tty or /dev/lp

You must answer the following queries when you use Quick Add to add a local printer to your system:

Printer type Select a printer type. A printer's type refers to a **/usr/src/cmd/terminfo** database entry that describes control codes the printer requires for handling various initialization and configuration operations as well as printing attributes. You choose one from a list generated from the **terminfo** database.

If you customize or add a new **terminfo** entry for a printer, include the source code in the **/usr/src/cmd/terminfo** directory and compile the source with the **tic(1M)** command to create a compiled version for the **/usr/lib/terminfo/...** directory.

Input types Select one type, also called content type, for every printer you add. An input type is a designation telling what kind of files you can print on the printer. We do not recommend specifying more than one content type. You choose from a list generated from the **/etc/lp/sysadm/input_types** file.

Device	Enter a device pathname. The pathname, indicating an entry in the <code>/dev</code> directory, must already exist.
Description	This is a comment field for entering any information you wish to record about the printer. The field may contain no more than 40 characters.

Quick Add for a local printer assumes the following defaults:

Table 7-1 Default attributes for local printers

Printer attribute	Default value
Interface script	standard
Stty options	clocal
Input options	nobanner
Device is also a login	no
Fault recovery	continue
Alert messages	mail
Alert interval	0
Users to allow	all
Users to deny	N/A
Forms to allow	none
Forms to deny	N/A
Enable printer	yes
Accept printer	yes

Remote lpNet to DG/UX 5.4 system and Remote lpNet to lpd system

You must answer the following queries when you use Quick Add to add an lpNet connected remote printer to your system:

Remote host	Enter the name of the remote system to which the printer is attached. See the lpssystem(1M) manual page, Chapter 4, and Chapter 8 for more information on the name.
Remote printer name	Enter the name of the printer on the remote system.
Description	This is a comment field for entering any information you wish to record about the remote printer. The field may contain no more than 40 characters.

Quick Add for a remote lpNet printer assumes the following defaults:

Table 7-2 Default attributes for remote printers connected via lpNet

Printer attribute	Default value
Printer type	printer-80
Input types	simple
Users to allow	all
Users to deny	N/A
Forms to allow	none
Forms to deny	N/A
Enable printer	yes
Accept printer	yes
Interface script	N/A
Stty options	N/A
Print options	N/A
Device is a login	N/A
Fault recovery	N/A
Alert messages	N/A
Alert interval	N/A

Remote to system other than DG/UX 5.4

You must answer the following queries when you use Quick Add to add a remote printer to your system from a non-DG/UX system or pre-5.4 system using **remshlp**:

Printer type Select a printer type. A printer's type refers to a **/usr/src/cmd/terminfo** database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.

For a generic 80-column line printer, choose **printer-80**. For a PostScript printer connected to a parallel port, termserver, or a MiLAN Fastport, select **PS-b** for batch mode. For a PostScript printer connected directly to a serial port, choose **PS**.

- Input types** Select one input type, also called content type, for every printer you add. An input type is a designation telling what kind of files you can print on the printer. We do not recommend specifying more than one content type.
- For a generic line printer, choose **simple**. For a PostScript printer, choose only **PS**.
- Interface scripts** Select the remote interface script, which is the interface script the LP service uses to establish the connection to the remote printer . You choose from a list generated from the **/etc/lp/sysadm/remote_models** file. The provided scripts all use the **Remsh** model.
- Remote host** Enter the name of the remote system to which the printer is attached. See the **lpssystem(1M)** manual page, Chapter 4, and Chapter 8 for more information on the name.
- Remote printer name** Enter the name of the printer on the remote system.
- Description** This is a comment field for entering any information you wish to record about the remote printer. The field may contain no more than 40 characters.

Quick Add for a remote printer assumes the following defaults:

Table 7-3 Default attributes for remote remsh printers

Printer attribute	Default value
Stty options	none
Print options	nobanner
Fault recovery	continue
Alert messages	mail
Alert interval	0
Forms to allow	none
Forms to deny	N/A
Users to allow	all
User to deny	N/A
Enable printer	yes
Accept printer	yes
Device	N/A
Device is a login	N/A

Remote networked printer device

You must answer the following queries when you use Quick Add to add a networked printer, such as a termserver or MiLAN Fastport, to your system:

Printer type Select a printer type. A printer's type refers to a `/usr/src/cmd/terminfo` database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.

For a generic 80-column line printer, choose **printer-80**. For a PostScript printer connected to a parallel port, termserver, or a MiLAN Fastport, select **PS-b** for batch mode. For a PostScript printer connected directly to a serial port, choose **PS**.

Input types Select one input type, also called content type, for every printer you add. An input type is a designation telling what kind of files you can print on the printer. We do not recommend specifying more than one content type.

For a generic line printer, choose **simple**. For a PostScript printer, choose **PS**.

Interface scripts	Select the network interface script, which is the interface script the LP service uses to establish the connection with the remote printer. You choose from a list generated from the <code>/etc/lp/sysadm/network_models</code> file.
Port number or name	Optionally, you may enter the port number or name for the networked printer device. You can get the name from <code>/etc/hosts</code> . The default port number is decimal 23.
Description	This is a comment field for entering any information you wish to record about the networked printer. The field may contain no more than 40 characters.

Quick Add for a networked printer assumes the following defaults:

Table 7-4 Default attributes for networked printers

Printer attribute	Default value
Stty options	none
Print options	nobanner
Device is a login	N/A
Fault recovery	continue
Alert messages	mail
Alert interval	0
Users to allow	all
Users to deny	N/A
Forms to allow	N/A
Forms to deny	N/A
Enable printer	yes
Accept printer	yes
Device	N/A

Full Add —

The Full Add operation gives you more control over the characteristics of the printer you are adding than the Quick Add operation. However, Full Add requires you to specify each of a printer's characteristics which can be somewhat lengthy. If you want to get a printer up and running quickly, you should use the Quick Add operation.

As with the Quick Add operation, the queries for adding a printer vary with the connection type of the new printer. The following sections discuss the queries for each connection type.

Local /dev/tty or /dev/lp

You must answer the following queries when you use Full Add to add a local printer to your system:

Printer type A printer's type refers to a **terminfo** database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.

Initially, the default printer type is **printer-80**. Although you may leave the printer's type set to the default, you can enhance your printer's ability to serve users by assigning it a more appropriate type. For a list of valid printer types, see Table 1-1.

The **terminfo** entries for emulated printer types generally have names of the form *printer-emulation*, where *printer* is the model of printer and *emulation* is the type of printer that it emulates. For example, **dg6617-epsonfx** is a Data General model 6617 printer emulating an Epson printer.

Input types allowed

You may set one or more input types, also called content types, for every printer you add. An input type is a designation telling what kind of files you can print on the printer. We do not recommend specifying more than one content type. Initially, the default input type is **simple**, which means common ASCII character text. PostScript printers are restricted to type **PS** only.

Input types also affect which filters are used. See the "Managing Filters" section later in this chapter for more information.

Generally, most printers can print **simple** files and files whose input type is the same as the printer type. Additionally, you can make up other input types to correspond to printer types and the types of files you print. Here are some accepted input types:

cif Output of BSD **cifpb**
processor.

epsonfs	Epson and compatible printers.
proprinter	IBM ProPrinter™ and compatible printers.
fortram	ASA carriage control format.
laserjet	Hewlett-Packard LaserJet® and compatible printers.
otroff	CAT typesetter instructions generated by BSD (old) troff .
plot	Plotting instructions for Tektronix displays and devices.
PS	PostScript language.
raster	Raster bitmap format for Varian raster devices.
simple	ASCII file.
tex	DVI format files.
troff	Device-independent output from the troff text-formatting processor.

Once you have assigned input types to your printers, users may submit requests specifying the input type with the **lp** command's **-T** option. As long as the request does not specify a destination and there is no default printer, the LP service will print the request on any printer that accepts the specified type. If a default printer has been set and you want to override it, specify **-d any** on the **lp** command line.

For example, the following command line overrides the system default printer and submits the PostScript file **myfile** to any printer that accepts input type **PS**:

```
% lp -d any -T PS myfile ↵
```

Note that **myfile** must already be in PostScript format. In general, when using the **lp** option **-T type**, the file should be in the specified *type* already.

Interface script Enter the interface script that the LP service uses to drive the printer. For a common line printer or laser printer not falling into one of the categories below, select the **standard** model. Other models are:

dg455x	For Data General text-only laser printers, models 4557 and 4558, or the replacement model 6454.
dg455x	For Data General text-only laser printers, models 4557 and 4558, or the replacement model 6454.
fastport	For networked printers using the Milan network printer box.
remshlp	For remote printers using a System V LP scheduler (the DG/UX default scheduler).
remshlp_bsd	For remote printers using a BSD spooler.
remshlp_a	For remote printers on an AOS/VS system.
termprinter_3com33	supports CS/2000 and 2100 Series TermServers.
termprinter_tcp	supports CS/3000 and 3100 Series TermServers.
termprinter_telnet	supports printers connected to a specific telnet port.
dg_annex	supports both the Annex and MicroAnnex Termservers. This is the default.
datability	supports the Datability VCP300/1000 TCP Series Termserver.
standard	For local line and PostScript printers.

You may copy these interface scripts, located in `/var/spool/lp/model` (links to `/usr/lib/lp/model`), and tailor them to fit printers on your system. For more information about interface scripts, see Chapter 9 “Administering the LP subsystem.”

Device You must supply a device pathname when adding a local printer. The pathname, indicating an entry in the `/dev` directory, must already exist.

The device pathname represents the port to which you have connected the printer. If your computer hardware has a dedicated line printer parallel port, this pathname is **/dev/lp**.

IMPORTANT: If you have connected your printer to an asynchronous terminal line, you should refer to your installation documentation and/or your controller documentation to ascertain which line it is. An asynchronous terminal pathname has the form **/dev/tty nn** , where nn is the line number.

IMPORTANT: If you have connected your printer to an asynchronous line, do not assign a port monitor service to that line. Use the Ports menu operations, making sure any service for the line is disabled.

Stty options Specify additional **stty** options in this query. **Stty** options control line I/O characteristics such as baud, flow control, parity, output processing, and so on. For binary printing, use the **stty** option **-opost**. For more information, see the **stty(1)** manual page.

Print options You may specify several print options to determine how the printer spaces jobs on the page. These options are:

length This option determines the number of lines printed per page. For example, to set page length to 63 lines, include the following option: **length=63**.

width This option determines the width of the printable area on the page. Specify the width in characters. For example, to set page width to 40 characters, include the following option: **width=40**.

cpi This option determines the horizontal printing density, specified in characters-per-inch. For example, to set characters-per-inch to 10, specify the following option: **cpi=10**.

lpi This option determines the vertical printing density, specified in lines-per-inch. For example, to print 6 lines per inch, include the following option: **lpi=6**.

nobanner/banneroff/banner

The banner is the job header page. **nobanner**, the default in **sysadm**, means print a banner before every job, but allow users to specify that no banner be printed. **banneroff** means never print a banner page. **banner** means always print a banner page.

When specifying these attributes for the Print Options query, list them separated by spaces, for example:

length=63 lpi=6

To set any of these printing attributes to their default values, specify them without a number, for example:

cpi=

In the case of the **banner** attribute, reset the default by specifying **banner**.

Device is also a login terminal

In some cases, the device used as a printing output device may also be used for logging into the system.

Fault recovery The LP service can handle a printer fault in one of three ways:

continue When the printer encounters a fault, it waits for the fault to clear and resumes printing at the point the fault occurred.

beginning When the printer encounters a fault, it clears the fault condition and reprints the entire job.

wait When the printer encounters a fault, the LP service stops the job and places the printer in a disabled state. The printer will not continue until you enable it. Enable a printer with the **sysadm** operation Device -> Printer -> Enable.

Alert messages When a printer fault occurs, the LP service can alert you to the condition in several different ways:

mail Using the **mail(1)** command, the LP service sends mail to the administrator.

write	Using the write(1) command, the LP service writes a message to the terminal where the administrator is currently logged in.
quiet	Send no alert for the current print fault. You can select this option only if the LP service has already determined that your printer is in a fault state. This option is useful for silencing alerts for a fault condition that you already know exists.
none	The LP service sends no alerts if it detects a fault.

As an alternative, you may write your own alert script. The script should accept the alert message as standard input from the LP service. When prompted, specify the command line for invoking your custom alert script.

You may also specify how often the LP service alerts you when a fault occurs. You specify the interval in minutes. For example, an interval of 10 means that the LP service will send out an alert every 10 minutes for as long as the fault condition exists. If you want the LP service to send out only one alert message per fault, specify 0.

Alert interval	When a printer fault occurs, the system sends periodic messages to alert the system administrator. The alert interval determines the number of minutes that the LP system will wait between messages. Specifying 0 indicates that you want only one message per fault sent to the administrator.
Users to allow	Optionally, you may specify which users may use the printer. Regardless of any users expressly denied access to the printer (via <code>Users to deny</code> , discussed below), only the users specified in <code>Users to allow</code> may use the printer. Specifying all allows access for all users except any who are denied access in <code>Users to deny</code> .
Users to deny	Optionally, you may specify which users may not use the printer. If you specify users in <code>Users to allow</code> , discussed previously, it is unnecessary to specify users in <code>Users to deny</code> .

Forms to allow	Specify forms, added with the Create operation of the Forms menu, that can be used on this printer. Regardless of any forms expressly restricted from the printer (via Forms to deny, discussed below), only the forms specified in Forms to allow may be used on the printer. Specifying all allows use of all forms except any that are restricted in Forms to deny.
Forms to deny	Specify forms that may not be used on this printer. If you specify forms in Forms to allow, discussed previously, it is unnecessary to specify forms in Forms to deny.
Enable	<p>Enabling the printer allows the LP service to send waiting jobs to it. If a printer is set to accept but it is disabled, the LP service accepts jobs submitted for the printer, but the LP service will not send the jobs to the printer for printing.</p> <p>The Device menu provides operations for enabling and disabling printers.</p>
Accept	<p>Setting the printer to accept requests allows users to submit jobs for that printer. If a user submits a job for a printer that is set to reject jobs, the lp command does not queue the job; instead, it returns a reject message to the user.</p> <p>The Device menu provides operations for setting printers to accept and reject job requests.</p>
Printer Description	This is a comment field for recording any information you wish to record about the printer. The field may contain no more than 40 characters.

Remote lpNet to DG/UX 5.4 system and Remote lpNet to lpd system

When adding a remote printer, make sure that the printer is physically attached to the remote system you specify. The DG/UX system does not allow you to send print requests through one remote system to reach a printer attached to yet another remote system.

You must answer the following queries when you use Full Add to add an **lpNet** connected remote printer to your system. For complete information about queries already covered, see the descriptions above for local printers.

Printer type	A printer's type refers to a terminfo database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.
Remote system	Enter the name of the remote system to which the printer is attached.
Remote printer name	Enter the name of the printer on the remote system.
Users to allow	Optionally, you may specify which users may use the printer.
Users to deny	Optionally, you may specify which users may not use the printer. If you specify users in <code>Users to allow</code> , discussed above, it is unnecessary to specify users in <code>Users to deny</code> .
Forms to allow	Specify forms, added with the <code>Create</code> operation of the <code>Forms</code> menu, that can be used on this printer.
Forms to deny	Specify forms that may not be used on this printer. If you specify forms in <code>Forms to allow</code> , discussed above, it is unnecessary to specify forms in <code>Forms to deny</code> .
Enable	Enabling the printer allows the LP service to send waiting jobs to it.
Accept	Setting the printer to accept requests allows users to submit jobs for that printer.
Printer Description	This is a comment field for recording any information you wish to record about the printer.

Remote to system other than DG/UX 5.4

When adding a remote printer, make sure that the printer is physically attached to the remote system you specify. The DG/UX system does not allow you to send print requests through one remote system to reach a printer attached to yet another remote system.

You must answer the following queries when you use `Full Add` to add a remote printer from a non-DG/UX system or pre-5.4 system using **remshlp** to your system. For complete information about queries already covered, see the descriptions above for local printers.

Printer type	A printer's type refers to a terminfo database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.
Input types allowed	Select one input type, also called content type, for every printer you add. We do not recommend specifying more than one content type. PostScript printers are restricted to PS only.
Interface script	Enter the interface script that the LP service uses to drive the printer.
Remote system	Enter the name of the remote system to which the printer is attached. See the lpssystem(1M) manual page, Chapter 4, and Chapter 8 for more information on the name.
Remote printer name	Enter the name of the printer on the remote system.
Stty options	Specify additional stty options in this query. For binary printing, use the stty option -opost . For more information, see the stty(1) manual page.
Print options	You may specify several print options to determine how the printer spaces jobs on the page.
Fault recovery	Specify how the LP service handles a printer fault.
Alert messages	Specify how the LP service alerts you if there is a printer fault.
Alert interval	Specify the amount of time between alert messages sent when a printer fault occurs.
Users to allow	Optionally, you may specify which users may use the printer.
Users to deny	Optionally, you may specify which users may not use the printer. If you specify users in <code>Users to allow</code> , discussed above, it is not necessary to specify users in <code>Users to deny</code> .
Forms to allow	Specify forms, added with the <code>Create</code> operation of the <code>Forms</code> menu, that can be used on this printer.

Forms to deny	Specify forms that may not be used on this printer. If you specify forms in <code>Forms to allow</code> , discussed above, it is not necessary to specify forms in <code>Forms to deny</code> .
Enable	Enabling the printer allows the LP service to send waiting jobs to it.
Accept	Setting the printer to accept requests allows users to submit jobs for that printer.
Printer Description	This is a comment field for recording any information you wish to record about the printer.

Remote networked printer device

When adding a remote printer, make sure that the printer is physically attached to the remote system you specify. The DG/UX system does not allow you to send print requests through one remote system to reach a printer attached to yet another remote system.

You must answer the following queries when you use Full Add to add a networked printer, such as a termserver or MiLAN Fastport, to your system. For complete information about queries already covered, see the descriptions above for local printers.

Printer type	A printer's type refers to a terminfo database entry that describes what control codes the printer requires for handling various initialization and configuration operations as well as printing attributes.
Input types allowed	Select one input type, also called content type, for every printer you add. We do not recommend specifying more than one content type. PostScript printers are restricted to PS only.
Interface script	Enter the interface script that the LP service uses to drive the printer.
Networked printer device name	Enter the name of the networked printer device, such as from /etc/hosts .
Port number or name	Optionally, you may enter the port number or name for the networked printer device.
Stty options	Specify additional stty options in this query.
Print options	You may specify several print options to determine how the printer spaces jobs on the page.

Fault recovery	Specify how the LP service handles a printer fault.
Alert messages	Specify how the LP service alerts you if there is a printer fault.
Alert interval	Specify the amount of time between alert messages sent when a printer fault occurs.
Users to allow	Optionally, you may specify which users may use the printer.
Users to deny	Optionally, you may specify which users may not use the printer. If you specify users in <code>Users to allow</code> , discussed above, it is not necessary to specify users in <code>Users to deny</code> .
Forms to allow	Specify forms, added with the <code>Create</code> operation of the <code>Forms</code> menu, that can be used on this printer.
Forms to deny	Specify forms that may not be used on this printer. If you specify forms in <code>Forms to allow</code> , discussed above, it is not necessary to specify forms in <code>Forms to deny</code> .
Enable	Enabling the printer allows the LP service to send waiting jobs to it.
Accept	Setting the printer to accept requests allows users to submit jobs to that printer.
Printer Description	This is a comment field for recording any information you wish to record about the printer.

Setting the default printer

You can make one of your printers the default printer so that any job submitted without the destination (`-d destination`) option goes automatically to the default printer. The default printer may be either a local printer or a remote one.

Deleting printer devices

The `Delete` operation removes the files, directories, and LP service database entries supporting the specified printer. Deleting an entry for a remote printer has no effect on the files, directories, and LP service database entries supporting the printer on the remote system. Deleting the last printer of a class also removes the class.

Modifying printer devices

Use the `Modify` operation to change the attributes that you set for a printer when you added it. For a complete discussion of the various

attributes, see “Adding printer devices.” The Modify operation for printers uses the same queries as a Full Add.

The Modify operation does not allow you to change the class to which a device belongs. To change the class membership, use the Modify operation of the Classes menu.

Accepting or rejecting requests

Use the Accept operation to allow users to submit requests to a printer device. To make the LP service reject requests for a given printer, use the Reject operation.

When a user submits a request to a printer that is rejecting requests, the LP service returns an error.

Setting a printer to reject requests does not remove jobs for that printer that are already in the print queue.

The accept or reject status of a printer has no effect on its enable or disable status, described below.

Enabling and disabling printer devices

The Enable operation makes a printer available to print requests. If the fault recovery attribute for a printer is set to “wait” (see “Adding printer devices”), you will have to enable a printer after a fault occurs.

The Disable operation makes a printer unavailable to print jobs. Users may still submit jobs for printing by the printer even though it is disabled (assuming it is set to accept requests). Requests submitted for a disabled printer will remain in the print queue until you remove them, move them, or enable the printer.

The Disable operation allows you to select how to handle a job currently being printed by the printer. You may choose to interrupt the job, in which case the operation cancels the job, removing it from the queue, or you may choose to let the printer finish the request before entering the disabled state.

When you disable a printer, you may specify a reason in the form of a line of text of your own choosing, that tells why you have disabled the printer. When a user displays the status of the printer, the display includes the reason that you entered.

Displaying devices

Use the List operation to display information about any or all printers that you have added on your system. When you invoke the List operation, you may select either the `setup` or `state` listing.

The `setup` listing shows not only some information about the current state of the printer, but it also displays the settings of attributes that you defined for the printer when you created it with the `Add` operation. This listing includes:

- Forms mounted
- Content types
- Printer types
- Description
- Connection
- Interface
- On fault
- After fault
- Users allowed
- Forms allowed
- Banner
- Character sets
- Default pitch
- Default page size
- Default port settings

See “Adding printer devices” for a discussion of these attributes.

The `state` listing shows only information pertinent to the current state of the printer, whether it is enabled and accepting requests, and how many requests are in its queue. The display also includes the device pathname.

Managing classes

A printer class is a group of printers to which you may submit a print request for printing by any printer in the class. When you specify a class on the `lp` command line, the LP service prints the job on the first available printer in the class. The advantage to this system is that users do not need to investigate the print queue every time they submit a job in order to see which printer is currently available or which printer already has jobs queued up. The result is faster throughput for your users and more efficient utilization of your printers.

Another advantage of print classes is that they allow you to maximize use of favored printers. For example, if you have a fast line printer and a slow line printer, you can create a class where the fast line printer is first and the slow one is second. Thus, the LP service submits the job to the fast printer if both are available and to the slow printer only when the fast printer is already in use.

Creating classes

The `Create` operation of the `Classes` menu creates a new printer class and allows you to add printers to the class. When you create

the class, you must add at least one printer to it. The printers you add should already exist and can be only local printers. Add printers using the operation Device -> Printer -> Devices -> Add.

When you create a class, the order in which you list the printers determines the order in which the LP service checks them when assigning jobs. When you submit a job to the class, the LP service assigns the job to the first printer in the list that is available. A printer may belong to more than one class.

The Create operation also allows you to specify whether you want the class to accept jobs or reject jobs. For normal operation, specify Accept. If you do not want users to use the class at this time, specify Reject. Even if you set a class to reject requests, a user can still use a printer in the class by specifying the printer name on the **lp** command line (using the **-d** option).

To submit a job for printing by a certain class, use the **-d** option of the **lp** command. For example, to print the file **myfile** at the first available printer in class **class1**, use this command:

```
% lp -d class1 myfile ↵
```

Modifying classes

Use the Modify operation of the Classes menu to modify a printer class. The class must already exist.

The Modify operation allows you to add printers to the class or remove members from the class. If you add a member to the class, the printer must already exist. Add a printer with the operation Device -> Printer -> Devices -> Add.

To change the order of members in the class, you need to remove them and add them again in the desired order. The order of printers in the class determines the order in which the LP service checks them when assigning jobs. When you submit a job to the class, the LP service assigns the job to the first printer in the list that is available.

While you are adding or removing printers, you can also change the accept or reject status of the class. Even if you set a class to reject requests, a user can still use a printer in the class by specifying the printer name on the **lp** command line (using the **-d** option).

Removing classes

Use the Remove operation to delete a printer class from the LP service. You do not have to delete the printers from the class before removing the class.

Displaying classes

To display information about the classes on your system, select the List operation from the Classes menu. You may list the status of all classes on the system, or you may specify a class for listing. The operation lists the printers that are members of the classes.

Managing filters

Filters are programs that process files before printing them. This section covers the **sysadm** operations, located in the Filters menu of the Printer menu, that you use to manage LP service filters.

A filter can function in three ways:

- To convert a file from one format to another before printing.
- To handle the kind of special printing modes that a user can request with the **lp** command's **-y** option: two-sided printing, landscape printing, draft quality printing, and so on.
- To detect printer faults and notify the LP service.

When defining a filter, you not only name the program that functions as the filter, you also specify what content types the filter will accept as input and produce as output.

A content type refers to the formatting, codes, or conventions used in a file to describe its page layout or contents. The purpose of having content types is to allow the LP service a means of matching user-submitted print requests with compatible filters and printers. When a user submits a job and specifies a content type, the LP service attempts to match the file's content type with the printer type or input type specified for a printer. If no satisfactory printer exists, the LP service looks for filters that can convert the file into a printable content type.

For example, you may have created filters that accept or produce PostScript content. If you submit a file for printing and you specify that the file is of type PostScript, the LP service will look for a printer whose input type is PostScript. If the print service does not find one, it will attempt to assemble a series of filters that can take your PostScript file and convert it to a type that one of your printers can print.

You set a printer's input content type when you add it with the Add operation in the Devices menu. To change a printer's input types, use the operation `Device -> Printer -> Devices -> Modify`.

IMPORTANT: For remote printing, if your connection type is either **Remote lpNet to lpd system** or **Remote lpNet to DG/UX 5.4 system**, all

filtering takes place on the remote system. For the following connection types: **Local /dev/tty**, **Remote to system other than dg/ux 5.4**, and **Remote to networked printer device**, filtering takes place on the local system.

The Filters menu provides operations for:

- Adding filters.
- Modifying filters.
- Removing filters.
- Restoring filters.
- Displaying filters.

The following sections elaborate on these operations in more detail. For information on writing a filter, see Chapter 9.

Adding filters

Use the Add operation to introduce a filter program to the LP service. When invoking the Add operation, be prepared to supply the following information:

- The command line for invoking the filter.
- The existing filter, if any, to copy to make the new filter.
- The content types the filter will accept as input.
- The content types the filter will produce as output.
- The printer types compatible with the output types.
- The printers that may print the filter's output.
- The filter speed.
- The options to use when invoking the filter.

The following sections elaborate.

Command line —

This is the full pathname of the filter program. If there are any fixed options that the program always needs, include them here.

Filter to copy —

This is any existing filter that you may want to copy to make the new filter.

Input types —

maybe include filter input types as shipped--Device input types?

This is the list of file content types that the filter can process. The content type designations are names that you make up based on the types of files your users print and the types of printers you have. The LP service matches file content types and filter output types with filter input types, so be careful to use consistent naming conventions. Most filters can accept only one content type as input.

Output types —

This is the list of file content types that the filter can produce. The LP services matches content output types with filter input types and printer input types, so, again, use consistent naming conventions. Most filters can produce only one content type as output.

Printer types —

This is the list of printer types for which the filter may produce output. In most cases, this list will be the same as the Output Types list. Any listed printer types should match types of existing printer devices.

Printers—

There may be printers who are compatible with the filter's output type but for other reasons are undesirable as output devices for this filter. When this case is true, you should specify the desirable printers in the Printers list, omitting the undesirable ones of the same type.

Filter speed —

You can designate a filter as either fast or slow. When a print request requires a filter designated "fast," the print service assigns a printer to the request at the same time it starts the filter. If a filter is particularly slow, however, this implementation occupies the printer unnecessarily because the printer is out of use while it waits for the filter to finish.

To avoid this kind of waste of printer time, you can designate such filters as "slow." The LP service executes slow filters in the background without causing the printer to wait; the slow filter does not access the printer (or the next filter in the series) until it has filtered the entire request. While the slow filter works, other requests are free to print.

If you are adding a filter that is intended to detect printer faults, you must designate the filter as "fast." You may designate a filter as "slow" if it does not require access to the printer.

Options —

The Options field allows you to specify how user options, LP service database settings, and preceding filter options can determine options to be passed to this filter.

For example, you can set the options field so that if a specified character set is in effect for this print request, the LP service should include a particular option on the filter command line.

Specifically, the Options field allows you to determine the filter invocation command line based on these printing attributes:

- Input content type.
- Output content type.
- Printer type.
- Printer name.
- Character pitch (characters per inch).
- Line pitch (lines per inch).
- Page length.
- Page width.
- Pages to print.
- Character set.
- Form name.
- Number of copies.
- Special modes.

For a detailed discussion on how to set the Options field for these attributes, see Chapter 9.

Modifying Filters

Use the Modify operation to change the definition for an existing filter. For a discussion of the features that make up a filter, see “Adding Filters” above.

Removing Filters

Use the Remove operation to delete the LP service database entries that support a given filter. The operation does not remove the filter program itself.

Restoring an Original Filter Definition

Use the Restore operation to restore a filter’s definition to the definition that originally shipped with the system. This operation functions only for filters that shipped with the DG/UX system.

Displaying Filters

This operation displays the attributes of filters currently available on your system. These attributes include:

Filter command name	The command name that invokes the filter.
Input content type	The list of content types that the filter accepts as input.
Output content type	The list of content types that the filter produces as output.
Printer type	The list of printer types for which the filter produces output.
Printer names	The list of printers for which the filter produces output.
Speed	The filter's speed.
Command	Any options that the LP service passes to the filter.

The "Adding Filters" section, earlier in this chapter, discusses these attributes in more detail.

Managing Forms

A form is a description of a page layout, plus some other attributes, that determine how a printer loaded with pre-printed form stock should complete special requests. For example, if you have a printer loaded with pre-printed purchase order forms, the form description provides the printer with descriptive information, such as lines per inch, characters per line, required print wheels, ribbon colors, and so on. The form description can also include an alignment pattern that you can print to make sure that the form stock is aligned correctly in the printer.

Once you have defined a form, you associate it with a printer by mounting it on that printer. The LP service then restricts use of the printer to printing requests that require that form. If a user submits a job requiring a form that is not mounted on the desired printer, the system alerts you with a message. You can determine where the alert message goes (who receives it) when you define the form.

To submit a job for a particular form, include the **-f** option on the **lp** command line. For example, if you have a file called **PO-3992** that is formatted to be printed as a purchase order using a form you call **porder**, print the file on the form with a command line like this:

```
% lp -f porder PO-3992 ↵
```

The Forms menu offers several operations for managing forms:

Create	Create a new form definition.
Modify	Change an existing form definition.
Remove	Delete a form description from the LP services databases.
Mount	Assign a form restriction to a printer.
Unmount	Remove a form restriction from a printer.
List	Display information about the forms currently defined on the system.
Show Status	Display the current status of forms on the system.

The following sections elaborate.

Creating Forms

Select the Create operation to enter a forms description. The operation asks a number of questions having to do with the desired page layout and other printing attributes. Some of these attributes, characters per line, for example, depend on the capabilities of your printer. You may need to consult your printer documentation to see what it can do and what it cannot do.

The following sections discuss the information you may specify when you define the form. The default form contains values for printing a normal page of 66x80 ASCII characters.

Form name —

The form name is any arbitrary name that you may choose. You should select a name that describes the form sufficiently but is still easy to type.

Form to copy —

When defining a form, you may specify an existing form whose definition you wish to start with as a model. If you need to define numerous forms that are similar to each other but that do not use many of the **sysadm**'s preset form defaults, you can create your own default form and copy the others from it.

Page length —

This value specifies the length of the form. For multi-page forms, this value is the length of each page. Specify the length as a

number of lines, or in inches or centimeters. To specify a length in inches or centimeters, follow the value with an **i** or **c**, as appropriate. For example, specify six inches as **6i**.

Page width—

This value specifies the width of the form. Specify the length as a number of characters, or in inches or centimeters. To specify a length in inches or centimeters, follow the value with an **i** or **c**, as appropriate. For example, specify 20 centimeters as **20c**.

Lines per inch/cm —

This value determines how many lines appear per vertical inch. You may specify this value in either lines per inch or lines per centimeter by following the value with an **i** or a **c**, respectively. If you omit the letter, the operation assumes that you mean inches.

Characters per inch/cm —

This value specifies the number of characters printed per horizontal inch. You may specify this value in either characters per inch or characters per centimeter by following the value with an **i** or a **c**, respectively. For example, specify 3 characters per centimeter with **3c**. If you omit the letter, the operation assumes that you mean inches.

Print wheel —

For this field, you may make up a name to represent a particular print wheel, character set, or font cartridge required to print the form. Make the name descriptive but easy to type. You should be consistent in your naming conventions.

Users can submit print requests specifying a particular print wheel or character set by including the **-S** option on the **lp** command line. If the required print wheel or character set is not mounted on the destination printer, the LP service will hold the job and alert you that you need to mount the print wheel or character set.

Ribbon color —

If the form requires a particular color of ribbon, you specify it in this field. Whenever you mount the form, the LP service will remind you to load the correct color of ribbon. The LP service does not track ribbons; therefore, it will not alert you if the wrong ribbon is loaded on a printer. It is up to you make sure the correct ribbon is loaded.

Number of pages —

This value specifies the number of pages making up a multi-page form.

Comment —

The comment field is for you to enter any information you wish about the form. Users can display the comment for a form; therefore, it is useful for you to enter a comment describing the form, its purpose, and so on.

Alignment pattern —

The alignment pattern is any pattern of characters that you may print to see if the form stock is loaded correctly in the printer. For security reasons, the LP service only allows the superuser and itself to see the alignment pattern.

Alert messages —

When a user submits a request requiring a form that is not mounted on a printer, the LP service alerts you. The alert may occur several different ways:

mail	Using the mail(1) command, the LP service sends mail to the administrator.
write	Using the write(1) command, the LP service writes a message to the terminal where the administrator is currently logged in.
quiet	Send no alert for the current form need. This option is useful for silencing alerts for a form need that you already know exists. Selecting quiet does not change the current alert status if there is currently no form need.
none	The LP service sends no alerts if it detects a need for a form.

As an alternative, you may write your own alert script. The script should accept the alert message as standard input from the LP service. When prompted, specify the command line for invoking your custom alert script.

Alert interval —

You may also specify how often the LP service alerts you when a form need arises. You specify the interval in minutes. For example, an interval of **10** means that the LP service will send out an alert every 10 minutes for as long as the need exists. If you want the LP service to send out only one alert message upon detecting a need for a form, specify **0**.

Job threshold —

If you do not want to receive an alert message every time someone needs a form mounted, you can set a threshold determining how many requests needing a form will occur before the LP service alerts you. For example, setting the threshold to **5** means that the LP service will not alert you to mount a form until it has received 5 print requests needing a form.

Users to allow —

Optionally, you may specify which users may use the form. Regardless of any users expressly denied access to the form (via `Users to deny`, discussed below), only the users specified in `Users to allow` may use the form. Specifying **all** allows access for all users except any who are denied access in `Users to deny`.

Users to deny —

Optionally, you may specify which users may not use the form. If you specify users in `Users to allow`, discussed above, it is not necessary to specify users in `Users to deny`.

Modifying Forms

Use the `Modify` operation to change the description for an existing form. For a discussion of the fields in the form description, see “Creating Forms.”

Removing Forms

To remove the description of a form from your system, select the `Remove` operation.

Mounting and Unmounting Forms

The LP service tracks forms status by allowing you to mount and unmount forms whenever you load or unload, respectively, a form on a printer. After loading form stock on a printer device, select the `Mount` operation to tell the LP service and users that the form is now available. With the form mounted, the LP service will print requests requiring the form.

After unloading form stock from a printer, use the `Unmount` operation to tell the LP service and users that the form is no longer available. If a user submits a request that requires a form that is not mounted, the LP service holds the job in the queue without printing it, and it sends you an alert message. The nature of the alert message depends on the `Alert Messages` option that you selected when you added the form. For more information on these options, see “Creating Forms,” earlier in the chapter. To change the

way the LP service currently handles alert messages, select the **Modify operation**.

To see what forms are mounted on what printers on your system, select the **Show Status operation**.

Displaying Forms

The **List operation** displays the form description attributes that you set with the **Create operation**. These attributes are:

Name	The name that you gave the form. This line in the display also tells whether or not the form is available for you to use.
Page Length	The length of the page in lines, inches, or centimeters. Values in inches and centimeters are followed by i or c , respectively.
Page Width	The width of the page in characters, inches, or centimeters. Values in inches and centimeters are followed by i or c , respectively.
Number of Pages	The number of pages making up the form.
Line Pitch	The number of lines per vertical inch.
Character Pitch	The number of characters per horizontal inch.
Character Set	The character set, print wheel, or font cartridge required by the form.
Ribbon Color	The color of ribbon required for the form.
Comment	A comment that you supplied when you added the form. This comment may describe the form, its purpose, and so on.
Alignment	A pattern of characters that you can print to see if the form stock is loaded correctly in the printer. For security reasons, only the superuser and the LP service can access and display the alignment pattern.

For a more detailed discussion of these attributes, see “**Creating Forms**.”

Showing Status of Forms

Use the **Show Status operation** to see what forms on your system are currently mounted and available to you.

Setting User and Request Priorities

The LP service offers several different ways of controlling which jobs print first when multiple jobs are competing for printing

resources. This priority mechanism revolves around the concept of a priority level associated with each job. Very simply, the lower the priority level number, the higher the priority of the request. Conversely, requests with higher priority numbers have lower priority. The LP service prints jobs with lower priority level numbers first.

Using the `-q` option of the `lp` command, users can set priority numbers for the jobs they submit. Priorities range from 0 (high priority) to 39 (low priority). For example, to submit a request to print file `nicefile` at the lowest priority, use this command:

```
% lp -q 39 nicefile ↵
```

To help regulate the use of the priority system, the LP service allows you to set limits for priorities that users can specify. The Priorities menu provides these operations:

Job Default	Set a default priority for requests that do not specify a priority with the <code>-q</code> option.
Remove	Remove a default job priority limit for a specific user.
Set	Set a default job priority limit for a specific user.
User Default	Set a default job priority limit for all users who do not have an individual limit.
List	List current job priority limits for general requests and for specific users.

The following sections describe these operations in more detail.

Job Default

Use the Job Default operation to set the priority for all requests submitted without the `-q` option. You use the `-q` option on the `lp` command line to set the priority for a job.

Remove

Use this operation to remove the priority limit set for an individual user. You set an individual limit with the Set operation. Once you have removed the individual limit, the user's requests are subject to the User Default, if set.

Set

This operation sets a priority limit for an individual user. The lower the priority level, the higher priority job they may submit.

Regardless of a user's priority limit, if the user does not specify a priority with the **-q** option, the request takes the system default priority level.

User Default

Use this operation to set a priority limit for all users for whom you have not set an individual limit. Regardless of the user default priority limit, if a user does not specify a priority with the **-q** option, the request takes the system default priority level.

List

This operation displays the priority values set with the other Priorities menu operations.

Managing Requests

The Requests menu offers several operations for handling jobs in the print queue. Unlike other printer management operations (other than list operations), some of these operations are accessible to users other than the superuser. The following sections elaborate.

Canceling a Request

Use the Cancel operation to remove a request from a queue. The superuser can cancel any request, but other users can cancel only their own requests.

Holding a Request

Any user can use the Hold operation to suspend his or her own request. A held request remains in the queue, but the LP service will not send it to a printer until the user (or superuser) releases the request with the Resume operation. A held request does not block the print queue: the LP service will continue to serve other requests.

Resuming a Held Request

Use the Resume operation to release a request that a user suspended with the Hold operation. Users other than the superuser can resume only their own requests.

Moving Requests

The superuser can use the Move operation to move requests from the queue for one printer or class to the queue for another printer or class.

Displaying Requests

Use the List operation to display the requests currently in the print queue. Like other List operations in **sysadm**, any user may use this operation.

Managing the Scheduler

The LP scheduler is the process that manages print queues and the other LP services. The system starts the LP scheduler when you bring the system up to run level 1. The Scheduler menu offers three operations, described in the next three subsections.

Starting the Scheduler

Use the Start operation to begin execution of the LP scheduler. This operation performs no function if the scheduler is already running.

Stopping the Scheduler

Use the Stop operation to halt execution of the LP scheduler. This operation performs no function if the scheduler is already stopped.

This operation does not remove jobs from the print queue. When you restart the scheduler, it resumes processing queued requests.

Displaying Scheduler Status

Use the List operation to display the status of the LP scheduler and the print queues. This operation tells whether or not the scheduler is running, and displays the queued requests for each printer on the system. For each request in a queue, this display shows:

- The request number.
- The user who submitted the request.
- The size of the file in bytes.
- The time and date when the job was submitted.
- Any printer specified.

Managing Remote Systems

When configuring remote printers using the lpNet connection method, you need to manage the remote systems. Use the operations in the Systems menu to manage the LP service databases that contain information about remote systems.

Before adding a remote printer that resides on a system using a DG/UX Release 5.4 LP scheduler, add an entry for the remote system using the Add operation in the Systems menu. On the remote system, the system administrator must do the same, adding similar information about your system. After adding this information on both systems, you can set up access to the remote system's printers, and the remote system administrator can set up access to your printers.

The following sections discuss the operations in the Systems menu.

Adding Remote Systems

Use this operation to add the required LP service database entries that allow you to share printing resources with a remote system. To add a remote system, you need to do several things:

Name	See the administrator of the remote system to verify the system's host name. If your systems are not part of an NIS (Network Information Services) domain, you need to make sure that entries for the host exist in your system's /etc/ethers and /etc/hosts files. To add entries to these files, see the sysadm utility's Networking menu. If you are in an NIS domain, the required entries may already exist in the NIS databases. If not, see your NIS administrator.
Scheduler Type	You need to know if the remote system's scheduler is compatible with the AT&T System V scheduler. If the remote system is an AViiON running DG/UX 5.4 or later, use the scheduler type s5 . Otherwise, use the BSD scheduler type (bsd).
Connection Timeout	Set a time in minutes that the connection with the remote system may remain idle before "timing out," or terminating.
Retry Period	Set the number of minutes to wait, after unexpected disconnection of service from the remote system, before attempting to reconnect.
Comment	Decide on an optional comment for the system entry. The List operation will display this comment with the other system information.

Modifying Remote Systems

Use the Modify operation to change the attributes associated with a system entry. See “Adding Systems” for information about the entry attributes.

Removing Remote Systems

The Remove operation deletes the remote system entry from the LP services databases. This operation does not remove entries from the */etc/ethers* or */etc/hosts* files.

Displaying Remote Systems

Use the List operation of the Systems menu to display remote system entries added with the Add operation. The section on the Add operation, above, describes the fields in the entry.

Displaying LP Service Status

The List operation of the Printer menu displays the status of the entire LP services subsystem. The display includes:

- Whether the scheduler is running.
- Which printer, if any, is the default destination.
- The device/printer name assignments.
- The accept status of each printer and the time and date when the accept status last changed.
- Whether the printer is enabled or disabled and the time and date when the enabled/disabled status last changed. If disabled, the report includes the reason.
- Which forms, if any, are available to you, and where they are mounted.
- Requests currently in the queue.

End of Chapter

8

Configuring printers from the shell

This chapter describes the shell-level commands that **sysadm** uses to provide LP services. Table 8–1 shows the **sysadm** menus and corresponding shell commands available for administering the LP service.

Table 8–1 LP print service menu and command summary

Task Description	Menu Selection	Shell Command
Configure printers for print service, set the default printer, turn queuing of requests on and off, enable and disable printers, display state and configuration of printers	Devices	lpadm (1M), accept (1M), reject (1M), enable (1M), disable (1), lpstat (1)
Group printers into classes	Classes	lpadm (1M)
Provide pre-processing software for files to be printed	Filters	lpfilter (1M)
Define pre-printed forms for print requests	Forms	lpforms (1M)
Define levels of priority available to users requesting print jobs	Priorities	lpusers (1M)
Cancel, hold, resume, move, and list print request	Requests	lp (1), cancel (1), lpmove (1M), lpsched (1M)
Start and stop print service, report status of scheduler and list print requests	Scheduler/Service	lpsched (1M), lpshut (1M), lpstat (1)
Set up communication to remote print service	Systems	lpystems (1M)
Identify active printers, print wheels & character sets, mounted forms, and pending requests	List	lpstat (1)

The rest of this chapter describes the work required to set up and maintain print services with the LP print service utilities. Details about the commands listed above are available in the manual pages for them.

This chapter includes:

- An overview of how the LP print service works
- Instructions for configuring the print service for your unique requirements
- Examples of how to configure printers
- Instructions on how to make printers available after configuring

Overview

The LP print service, originally called the LP spooler, is a set of software utilities that allows you, minimally, to send a file to be printed while you continue with other work. The term “spool” is an acronym for “simultaneous peripheral output on-line,” and “LP” originally stood for Line Printer, but has come to include many other types of printing devices. The print service has many optional enhancements; however, you can make your print service as simple or as sophisticated as you like.

Components of a print service

A print service consists of both hardware and software. You must have at least one computer and one printing device for an LP print service. Beyond that, you may have any number of computers and printing devices; there is no limit to the number of pieces of hardware you may include. The software consists of the LP print service utilities and any filters (programs that process the data in a file before it is printed) that you may provide. Users of your print service may be required to print all their files in the same format, or, if you make different types of printers and/or filters available with your service, they may choose from several formats. You may also offer your users a choice between plain paper and pre-printed forms (such as invoices or checks).

Functions performed by the print service software

Whether your print service is simple (such as a one-computer/one-printer configuration that prints every file in the same format on the same type of paper) or a sophisticated one (such as a computer network with multiple printers and a choice of printing formats and forms), the LP software helps you maintain it by performing several important functions:

- Scheduling the print requests of multiple users
- Scheduling the work of multiple printers

- Starting programs that interface with the printers
- Filtering users' files (if necessary) so they will be printed properly
- Keeping track of the status of jobs
- Keeping track of forms and print wheels currently mounted and alerting you to mount needed forms and print wheels
- Alerting you to printer problems

Suggestions for LP print service administration

Here are some tips about how to organize your duties as the administrator of an LP print service.

Configuring printer sites

Where you decide to put your printers and how you decide to connect them to your computers depends on how those printers will be used. There are three possible scenarios: (1) users may access printers attached to their own computer; (2) users may access printers attached to a server computer; and (3) users may access remote printers on a network to which their computer belongs.

- You may want to connect a particular printer directly to the computer that is the home system of the users who will use that printer most often. If you do, the type of connection you have will be referred to as a direct connection. An environment that includes more than one computer, each of which has a direct connection to a printer, is said to have a *distributed printing configuration*.
- You may want to have all your printers in one physical location, such as a computer center. If so, you might connect them all to one computer. Users on other computers who want to use a printer may access it through a network linking their own computers to the computer serving the printers. An environment in which one computer serves several printers (which can be accessed only through a computer-to-computer network) is described as a *print server configuration*.

Figure 8–1 shows a sample print server configuration.

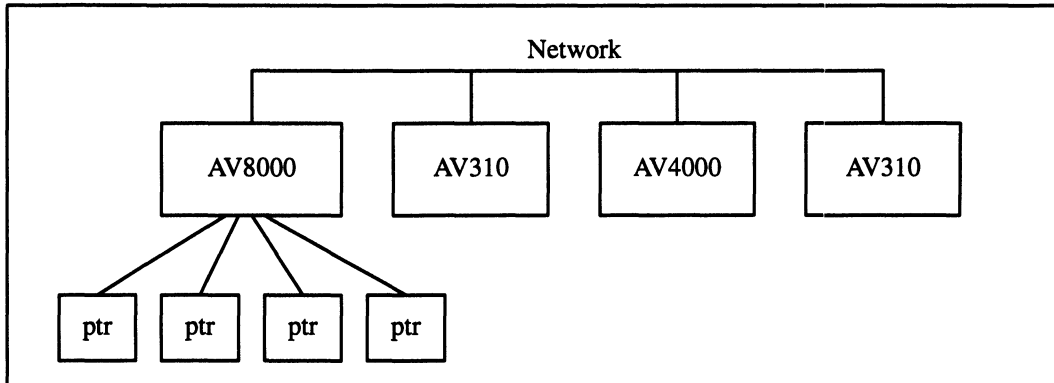


Figure 8–1 Print server configuration

You may want to link most of your printers to a dedicated printer server computer, while allowing other printers to be connected to your system. If so, you can arrange your computers and printers in a network configuration as shown in Figure 8–2.

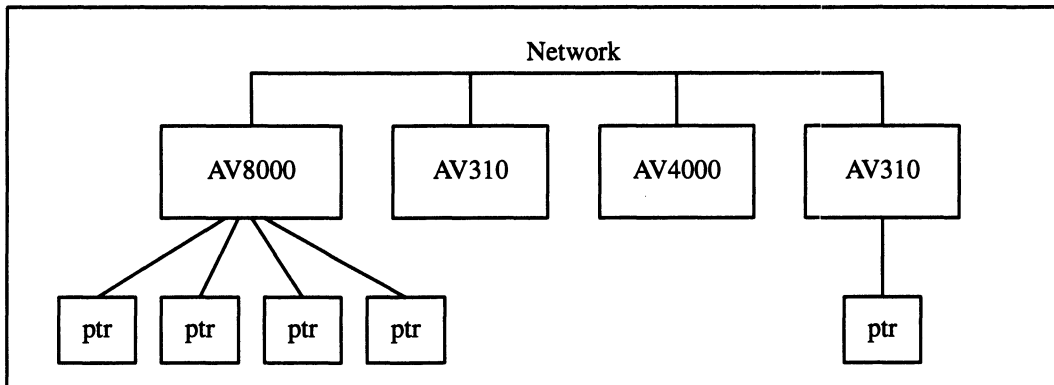


Figure 8–2 Network configuration

Configuring printers

Before the LP print service can start accepting print requests, you will have to describe the characteristics of each printer you have. The following is a list of the attributes most commonly defined:

- printer name (mandatory)
- connection method (mandatory for local printers)
- system name (mandatory for access to remote printers and mandatory for allowing remote access to local printers)
- interface program

- printer type
- content types
- printer port characteristics
- character sets or print wheels
- alerting to mount a print wheel
- forms allowed
- printer fault alerting
- printer fault recovery
- restrictions on user access
- inclusion of banner page in output
- printer description
- default printing attributes
- printer class membership
- system default destination
- mounting a form or print wheel
- removing a printer or class

You need to specify very little of this information to add a new printer to the LP print service. The more information you provide, however, the better the printer will satisfy various users' needs.

The descriptions in the sections below will help you understand what this printer configuration information means and how it is used, so that you can decide how to configure your printers. In each section you will also be shown how to specify this information when adding a printer. While you can follow each of the sections in order and correctly configure a printer in several steps, you may want to wait until you've read all the sections before adding a printer, so that you can do it in fewer steps.

Printer name

The printer name and the connection method (described next) are the only items you must specify to define a new local printer. To define a new remote printer, you must specify the printer name and the name of the remote system. The printer name is used to identify the printer, both by you (when you want to change the printer configuration or manage the printer), and by users who want to use the printer. The name may contain a maximum of 14 alphanumeric characters and underscores.

You may choose any name you like, but it is good practice to choose a name that is meaningful to the users of the LP print service. For example, **laser** is a good name for a laser printer. If you have several laser printers you may name them **laser1**, **laser2**, and so on.

You don't have to try to fit a lot of descriptive information into the name; there is a better place for this information (see "Printer description" below). You also don't have to make the name precisely identify the type of printer; users who need to use a particular type of printer can specify it by type rather than name (see "Printer type," below).

You will use the printer name every time you want to refer to the printer: when adding other configuration information for the printer, when changing the configuration of the printer, when referring to the status of the printer, and when removing the printer. Thus the first thing you must do to add a printer is identify its name. You will do this as shown below; but don't do it yet because you'll also need to specify the connection method.

```
# lpadmin -p printer-name ↵
```

There are no default names; you must name every printer.

Connection method

The LP print service lets you connect a printer to your computer through local and remote lpNet connections.

Figure 8-3 shows these connections.

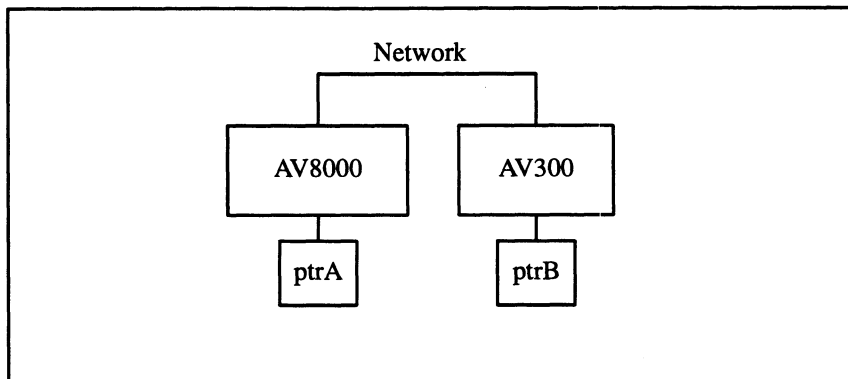


Figure 8-3 Methods of Connecting a Printer to a Computer

The AViiON 8000 system accesses printer **ptrA** through a local connection and accesses printer **ptrB** over the network. The AViiON 300 system accesses printer **ptrB** via local connection and printer **ptrA** over the network.

The simplest connection method is a local connection, which is to connect a printer directly to your computer. You may, however, want to connect a printer to a network (so it can be shared with other computers or workstations). Whichever method you use, you must describe it to the LP print service after you have connected the hardware.

To define the connection method for a new printer to your print service, invoke the **lpadmin** command, specifying a connection method through either the **-v** option for a locally connected printer or the **-U** option for a printer connected to a network.

The next two sections discuss local and remote connections. For information on network connections, see Chapter 6.

Local connections

The simplest and most common method by which printers are connected to a computer is local connection. If you use this method, you generally need to specify only two items on the command line when you make the connection: the name of the printer and the name of the connecting port. To connect a printer directly to your computer enter the following command:

```
# lpadmin -p printer-name -v pathname ↵
```

where *pathname* is the name of the special device file representing the printer port. The following are examples of typical names of special device files.

```
/dev/tty00 (serial)
/dev/tty16 (serial)
/dev/lp    (parallel)
```

(For details about using these files, see “Printer port characteristics” later in this chapter.)

Using a printer as a login terminal—

Some locally connected printers can also be used as terminals for login sessions. If you want to use a printer as a terminal, you must arrange for the LP print service to handle it as such. To do so, use the **-l** option to the **lpadmin** command, as follows:

```
# lpadmin -p printer-name -v pathname -l ↵
```

As before, *pathname* is the name of the special file representing the printer port. If the **-l** option is specified, the printer will be disabled automatically whenever the LP print service is started, and

therefore will have to be manually enabled before it can be used for printing. For instructions on manually enabling a printer, see “Enabling and disabling a printer” later in this chapter.

Remote lpNet connections

This section applies only if you are making a remote printer accessible to users on your system. A remote printer is one that is connected to a system other than your local system that you can access only through that remote system.

To make a remote system accessible, the name of the system on which the printer resides must be registered with the print service. If the remote printer resides on a DG/UX Release 5.4 system or on a System V Release 4 system, enter:

```
# lpsystem system-name ↵
```

If the remote printer resides on a BSD system, enter

```
# lpsystem -t bsd system-name ↵
```

IMPORTANT: If your system is using fully qualified names (for example, **shaman.hawk.rp.com**), then the full name must be used in the command line. If your network contains hosts using both simple (for example, **shaman**) and fully qualified names, then both the simple and full name should be added for each print server and client.

For details about the options available with this command, see the **lpsystem(1M)** manual page.

In either case, after entering the **lpsystem** command, enter the **lpadmin** command, as follows:

```
# lpadmin -p printer -s system-name ↵
```

where *printer* is the name by which your users identify the remote printer. You can usually use the same name used for that printer by the remote system. If the name used by the remote system is the same name used for an existing printer or class on your system, you must use a different name. To assign a different name to a remote printer, enter the following:

```
# lpadmin -p local-name -s system-name|remote-name ↵
```

For example, imagine you want your users to have access to a printer called **psjet2** that resides on a remote system called **newyork**. Because you already have a printer called **psjet2** on

your own system, you want to give the remote printer a new name on your system: **psjet3**. Request the new name by entering the following:

```
# lpadm -p psjet3 -s newyorklpsjet2 ↵
```

Before you add a remote printer to your system, be sure communications between your system and the network have been properly set up and verified.

Allowing remote users to access local printers

Making the printers on your local system accessible to users on remote systems is a two-step process: you must configure the port monitor on the local system, and you must register the remote system with the local LP print service. This section provides instructions for these tasks.

Configuring the local port monitor —

If a remote system requires access to printers connected directly to your system, you need to configure the local port monitor for the network you share to accept service requests and to notify the LP print service of such requests. For DG/UX Release 5.4 systems or System V systems calling your systems, issue the following command:

```
# pmadm -a -p netname -s lp -i root -v 'nlsadmin -V' -m 'nlsadmin \
-o /var/spool/lp/fifos/listenS5' ↵
```

where *netname* is the name of a network such as **tcp**.

If you expect users on BSD systems to send print requests to your system, then you need to configure your local port monitor. First, however, you need to know the Internet address of your system. To get this address in the correct hexadecimal format, run the **lpsystem** command with the **-A** option, as follows:

```
# lpsystem -A ↵
```

You then execute the following command, substituting the hexadecimal number for the argument address:

```
# pmadm -a -p tcp -s lpd -i root -v 'nlsadmin -V' -m 'nlsadmin \ -o
/var/spool/lp/fifo/listenBSD -A'address" ↵
```

Adding a system entry —

If you want your system to accept jobs from a remote system (and vice-versa), the print service must know about that system. The

lpssystem command allows you to register remote systems with the local print service. Run the command as follows:

```
# lpssystem system-name ↵
```

where *system-name* is the name of the remote system.

IMPORTANT: If your system is using fully qualified names (for example, **shaman.hawk.rp.com**), then the full name must be used in the command line. If your network contains hosts using both simple (for example, **shaman**) and fully qualified names, then both the simple and full name should be added for each print server and client.

If the remote host uses a pre-DG/UX Release 5.4 LP scheduler, you also need to add the host's Internet address to your **hosts** database. To add an entry to your **hosts** database, use the TCP/IP Databases menu of **sysadm**'s Networking menu.

Interface program

This section does not apply if you are making a remote printer accessible to users on your system. This is the program the LP print service uses to manage the printer each time a file is printed. It has several tasks:

- to initialize the printer port (the connection between the computer and the printer)
- to initialize the printer (restore it to a normal state in case a previously printed file has left it in an unusual state) and set the character pitch, line pitch, page size, and character set requested by the user
- to print a banner page
- to run a filter that prepares the file for printing
- to manage printer faults

If you do not choose an interface program, the standard one provided with the LP print service will be used. This should be sufficient for most of your printing needs. If you prefer, however, you can change it to suit your needs, or completely rewrite your own interface program, and then specify it when you add a new printer. For details on how to customize an interface program, see Chapter 9.

If you are using the **standard** interface program, you needn't specify it when adding a printer. If, however, you will be using a different interface program on a local printer, you can refer to it either by specifying its full pathname or by referring to another printer using the same interface program.

To identify a customized interface program by name, specify the printer name and the pathname of the interface program, as follows:

```
# lpadmin -p printer-name -i pathname ↵
```

To use a customized interface program of another printer, specify the printer names as follows:

```
# lpadmin -p printer-name1 -e printer-name2 ↵
```

Printer-name1 is the name of the printer you are adding;
printer-name2 is the name of an existing printer that is using the customized interface program.

Printer types

A printer type is the generic name for a printer. When you set up your system you can enhance its ability to serve your users by classifying, on the basis of type, the printers available through the print service. Assigning a type for each printer is also important because the LP software extracts information about printers from the **terminfo** database on the basis of type. This information includes the list of the printer's capabilities that is used to check the configuration information you supply to the print service. (By checking information provided by you against the capabilities of the printer, the print service can catch any inappropriate information you may have supplied.) The **terminfo** database also specifies the control data needed to initialize a particular printer before printing a file.

You can assign several types to a printer if the printer is capable of emulating more than one kind of printer. If you specify more than one printer type, the LP print service will use one of them as appropriate for each print request.

Although you do not need to specify a printer type, we recommend that you do so; when a printer type is specified, better print services can be provided.

To specify a printer type, use the following command line:

```
# lpadmin -p printer-name -T printer-type-list ↵
```

If you give a list of printer types, separate the names with commas. If you do not define a printer type, the default **unknown** will be used.

Content types

While the printer type tells the LP print service what types of printers are being added, content types (also called input types) tell

the LP print service what types of files can be printed directly on each printer. Most printers can print files of two types: the same type as the printer type (if the printer type is defined) and the type **simple**, (meaning an ASCII file) which is the default content type for all printers.

Some printers, though, can accept (and print properly) several different types of files. When adding this kind of printer, specify the names of the content types the new printer accepts by adding these names to the list. (By default, the list contains only one type: **simple**.) If you are adding a remote printer, enter the following command on the remote system to list the content types that it accepts:

```
# lpstat -p printer -l ↵
```

To specify the list of content types for the local printer, enter the following command:

```
# lpadmin -p printer-name -I content-type-list ↵
```

The *content-type-list* is a list of names separated by commas or spaces. If you use spaces to separate the names, enclose the entire list (but not the **-I**) in quotes.

Content type names may look a lot like printer type names, but you are free to choose names that are meaningful to you and the people using the printer. (The names **simple** and **any** are recognized as having particular meanings by the LP print service; be sure to use them consistently. The name **terminfo** is also reserved, as a reference to *all* types of printers.) The names must contain no more than fourteen characters and may include only letters, digits, and underscores. Table 8-2 describes some accepted content types.

Table 8-2 Content types

Types	Description
troff	Device independent output from troff
otroff	CAT typesetter instructions generated by BSD (old) troff
tex	DVI format files
plot	Plotting instructions for Tektronix displays and devices
raster	Raster bitmap format for Varian raster devices
cif	Output of BSD cifpbt
fortran	ASA carriage control format

Continued

Table 8-2 Content types

Types	Description
epsonfx	Epson and compatible printers
proprinter	IBM ProPrinter and compatible printers
laserjet	Hewlett-Packard LaserJet and compatible printers
PS	PostScript language
simple	ASCII file

When a file is submitted to the LP print service for printing with the printer specified by the **-d any** option of the **lp** command, the print service searches for a printer capable of handling the job. The print service can identify an appropriate printer through either the content type name or the printer type name. Therefore, you may specify either name (or no name) when submitting a file for printing. If the same content type is printable by several different types of printers, you should use the same content type names when you add those printers. This makes it easier for the people using the printers, because they can use the same name to identify the type of file they want printed regardless of the printing destination.

Most manufacturers produce printers that accept simple ASCII files. While these printers are different types (and thus have different initialization control sequences), they may all be capable of handling the same type of file, which we call **simple**. As another example, several manufacturers may produce printers that accept ANSI X3.64 defined escape sequences; however, the printers may not support all the ANSI capabilities; they may support different sets of capabilities. You may want to differentiate them by assigning different content types to these printers.

However, while it may be desirable (in situations such as these) to list content types for each printer, it is not always necessary to do so. If you don't, the printer type will be used as the name of the content type the printer can handle. If you have not specified a printer type, the LP print service will assume the printer can print only files of content type **simple**. This may be sufficient if you require users to specify the proper printer explicitly and if files are properly prepared for the printer before being submitted for printing.

The default content type: simple —

Files of content type **simple** are assumed to contain only two types of characters, printable ASCII characters and the following control characters:

backspace	moves the carriage back one space, except at the beginning of a line.
tab	moves the carriage to the next tab stop; by default, stops are spaced every 8 columns on most printers.
linefeed	moves the carriage to the beginning of the next line (may require special port settings for some printers—see “Printer port characteristics” below).
form feed	moves the carriage to the beginning of the next page.
carriage return	moves the carriage to the beginning of the same line (may fail on some printers).

The word “carriage” may be archaic for modern laser printers, but these printers do actions similar to those done by a carriage. If a printer can handle several types of files, including **simple**, you must include **simple** in the content type list; the type **simple** is not automatically added to any list you give. If you *don't* want a printer to accept files of type **simple**, give a blank *content-type-list*, as follows:

```
# lpadmin -p printer-name -I "" ↵
```

Printer port characteristics

This section applies only to local printers.

The interface program needs to set the characteristics of the port to which the printer is connected. These characteristics define the low level communications with the printer. Included are the baud rate; use of XON/XOFF flow control; 7, 8, or other bits per byte; type of parity; and output post-processing. The standard interface program will use the **stty** command to initialize the printer port, minimally setting the baud rate and a few other default characteristics.

The default characteristics applied by the standard interface program are listed below.

Table 8–3 Default printer port characteristics

Default	Meaning
9600	9600 baud rate
cs8	8-bit bytes
-cstopb	1 stop bit per byte
-parenb	no parity generation
ixon	enable XON/XOFF flow control
-ixany	allow only XON to restart output
opost	post-process data stream as listed below: -oluc don't map lower-case to upper-case onlcr map line feed into carriage-return -ocrnl don't map carriage-return into linefeed -onocr output carriage-returns even at column 0 nl0 no delay after linefeed cr0 no delay after carriage-return tab0 no delay after tab bs0 no delay after backspace vt0 no delay after vertical tab ff0 no delay after form-feed

IMPORTANT: If you are printing binary or graphics files, use the **stty** option **-opost** rather than the default **opost**.

You may find that the default characteristics are sufficient for your printers; however, printers vary enough that you are likely to find that you have to set different characteristics. For a complete list of characteristics, see the **stty(1)** manual page.

If you have a printer that requires printer port characteristics other than those handled by the **stty** program, you may have to customize the interface program. See Chapter 9 for more information. You can set the characters for a specific print request using the **lp** command option **-o stty=stty-option-list**. You can also use the **lpadmin** command option **-o stty=stty-option-list** to set the characteristics on a per print queue basis.

When you add a new printer, you may specify an additional list of port characteristics. The list you give will be applied after the default list, so that you do not need to include in your list items that you don't want to change. Specify the additional list as follows:

```
# lpadmin -p printer-name -o "stty=stty-option-list" ↵
```

Note that both the double quotes and single quotes are needed if you give more than one item in the *stty-option-list*.

As one example, suppose your printer is to be used for printing graphical data, where linefeed characters should be output alone, without an added carriage-return. You would enter the following command:

```
# lpadmin -p printer-name -o "stty=--onlcr" ↵
```

Note that the single quotes are omitted because there's only one item in the list.

As another example, suppose your printer requires odd parity for data sent to it. You would enter the following command:

```
# lpadmin -p printer-name -o "stty='parenb parodd cs7'" ↵
```

Character sets or print wheels

Although your users may use character sets or print wheels that have been mounted on a remote printer (by the administrator of the remote system on which that printer resides), you cannot mount a character set or a print wheel on a remote printer. Printers differ in the way they can print in different font styles. Some have changeable print wheels, some have changeable font cartridges, others have preprogrammed, selectable character sets.

When adding a printer, you may specify what print wheels, font cartridges, or character sets are available with the printer. If you're adding a remote printer and you want your users to be able to use character sets or print wheels that have been mounted by the administrator of the remote system, you must list those character sets and print wheels, just as you would list the character sets and print wheels on a local printer. (See instructions below.)

Only one of these is assumed to apply to each printer. From the point of view of the LP print service, however, print wheels and changeable font cartridges are the same because they require you to intervene and mount a new print wheel or font cartridge. Thus, for ease of discussion, only print wheels and character sets will be mentioned.

When you list the print wheels or character sets available, you will be assigning names to them. These names are for your convenience and the convenience of the users. Because different printers may have similar print wheels or character sets, you should use common names for all printers. This allows a user to submit a file for printing and ask for a particular font style, without regard for which printer will be used or whether a print wheel or selectable character set is used.

If the printer has mountable print wheels, you need only list their names. If the printer has selectable character sets, you need to list their names and map each one into a name or number that uniquely identifies it in the **terminfo** database. Use the following command to determine the names of the character sets listed in the **terminfo** database.

```
# tput -T printer-type csnm 0 ↵
```

Printer-type is the name of the printer type in question. The name of the character set obtained by default after the printer is initialized (0th character set) is printed. Repeat the command, using **1**, **2**, **3**, and so on in place of the **0**, to see the names of the other character sets. In general, the **terminfo** names should closely match the names used in the user documentation for the printer; however, because not all manufacturers use the same names, the **terminfo** names may differ from one printer type to the next. For the LP print service to be able to find the names in the **terminfo** database, you must specify a printer type. See “Printer type” above.

To specify a list of print wheel names when adding a printer, enter the following command.

```
# lpadmin -p printer-name -S print-wheel-list ↵
```

The *print-wheel-list* is a comma or space separated list of names. If you use spaces to separate the names, enclose the entire list (but not the **-S**) in quotes.

To specify a list of character set names and to map them into **terminfo** names or numbers, enter the following command:

```
# lpadmin -p printer-name -S character-set-list ↵
```

The *character-set-list* is also a comma or space separated list; however, each item in the list looks like one of the following:

```
csN=character-set-name  
character-set-name1=character-set-name2
```

The *N* in the first case is a number from 0 to 63 that identifies the number of the character set in the **terminfo** database. The *character-set-name1* in the second case identifies the character set by its **terminfo** name. In either case the name to the right of the equal sign (=) is the name you may use as an alias of the character set. You do not have to provide a list of aliases for the character sets if the **terminfo** names are adequate. You may refer to a character set by number, by **terminfo** name, or by your alias.

For example, suppose your printer has two selectable character sets (sets #1 and #2) in addition to the standard character set (set #0). The printer type is **5310**. You enter the following commands to determine the names of the selectable character sets.

```
% tput -T 5310 csnm 1 ↵  
english  
% tput -T 5310 csnm 2 ↵  
finnish
```

The words `english` and `finnish`, the output of the commands, are the names of the selectable character sets. You feel that the name `finnish` is adequate for referring to character set #2, but better names are needed for the standard set (set #0) and set #1. You enter the following command to define synonyms.

```
# lpadmin -p printer-name -S "cs0=american, english=british" ↵
```

The following three commands will then produce identical results.

```
# lp -S cs1 -d any ... ↵  
  
# lp -S english -d any ... ↵  
  
# lp -S british -d any ... ↵
```

If you do not list the print wheels or character sets that can be used with a printer, then the LP print service will assume the following: a printer that takes print wheels has only a single, fixed print wheel, and users may not ask for a special print wheel when using the printer; and a printer that has selectable character sets can take any `csN` name or `terminfo` name known for the printer.

Alerting to mount a print wheel

This section does not apply if you are making a remote printer available to users on your system.

If you have printers that can take changeable print wheels, and have listed the print wheels allowed on each, then users will be able to submit a print request to use a particular print wheel. Until it is mounted though (see “Mounting a form or print wheel” later in this chapter), a request for a print wheel will stay queued and will not be printed. You could periodically monitor the number of print requests pending for a particular print wheel, but the LP print service provides an easier way: you can ask to be alerted when the number of requests waiting for a print wheel has exceeded a specified threshold.

You can choose one of several ways to receive an alert.

- You can receive an alert via electronic mail. For more information on the mail system, see the **mailx(1)** manual page.
- You can receive an alert written to any terminal on which you are logged in. See the manual page for the **write(1)** command.
- You can receive an alert through a program of your choice.
- You can receive no alerts.

If you elect to receive no alerts, you are responsible for checking to see whether any print requests haven't printed because the proper print wheel isn't mounted. In addition to the method of alerting, you can also set the number of requests that must be queued before you are alerted, and you can arrange for repeated alerts every few minutes until the print wheel is mounted. You can choose the rate of repeated alerts, or you can opt to receive only one alert for each print wheel.

To arrange for alerting to the need to mount a print wheel or character set, enter one of the following commands:

```
# lpadmin -S print-wheel-names -A mail -Q requests -W minutes ↵
# lpadmin -S print-wheel-names -A write -Q requests -W minutes ↵
# lpadmin -S print-wheel-names -A 'command' -Q requests -W minutes ↵
```

The **print-wheel-names** argument may be a space- or comma-separated list of print wheels or character sets. The first two commands direct the LP print service to send you a mail message or write the message directly to your terminal, respectively, for each alert. The third command directs the LP print service to run the *command* for each alert. The shell environment currently in effect when you enter the third command is saved and restored for the execution of *command*; this includes the environment variables, user and group IDs, and current directory. The argument *requests* is the number of requests that need to be waiting for the print wheel before the alert is triggered, and the argument *minutes* is the number of minutes between repeated alerts.

If you do not want the print service to issue an alert when a print wheel needs to be mounted, enter the following:

```
# lpadmin -S print-wheel-names -A none ↵
```

If you want mail sent or a message written to another user when a printer fault occurs, use the third command with the option **-A'mail username'** or **-A'write username'**.

When you start receiving repeated alerts, you can direct the LP print service to stop sending you alerts (for the current case only), by executing the following command.

```
# lpadmin -S print-wheel-names -A quiet ↵
```

Once the print wheel has been mounted and unmounted again, alerts will start again if too many requests are waiting. Alerts will also start again if the number of requests waiting falls below the **-Q** threshold and then rises up to the **-Q** threshold again, as when waiting requests are canceled, or if the type of alerting is changed.

If *print-wheel-name* is **all** in any of the commands above, the alerting condition will apply to all print wheels for which an alert has already been defined.

If you don't define an alert method for a print wheel, you will not receive an alert to mount it. If you do define a method without the **-W** option, you will be alerted once for each occasion.

Forms allowed

For information about how to define, mount, and set up alerting to mount a form, see Chapter 9.

You can control the use of preprinted forms on any printer, including remote printers. (Although you cannot mount forms on remote printers, your users may use forms on remote printers—if the form is defined exactly the same on the local and remote systems.) You may want to do this, for instance, if a printer is not well suited for printing on a particular form because of low print quality, or if the form cannot be lined up properly in a local printer.

The LP print service will use a list of forms allowed or denied on a printer to warn you against mounting a form that is not allowed on the printer. However, you have the final word on this; the LP print service will not reject the mounting. The LP print service will, however, reject a user's request to print a file on a printer using a form not allowed on that printer. If, however, the printer is a local printer and the requested form is already mounted, the request will be printed on that form.

If you try to allow a form for a printer, but the printer does not have sufficient capabilities to handle the form, the command will be rejected.

The method of listing the forms allowed or denied for a printer is similar to the method used to list users allowed or denied access to the **cron** and **at** facilities. See the manual page for **cron(1M)** and **crontab(1)**. Briefly, the rules are as follows:

- An allow list is a list of forms that are allowed to be used on the printer. A deny list is a list of forms that are not allowed to be used on the printer.
- If the allow list is not empty, only the forms listed are allowed; the deny list is ignored. If the allow list is empty, the forms listed in the deny list are not allowed. If both lists are empty, there are no restrictions on which forms may be used other than those restrictions that apply to a printer of a particular type, such as a PostScript printer for which a license is required.
- Specifying **all** in the allow list allows all forms; specifying **all** in the deny list denies all forms.

You can add names of forms to either list using one of the following commands:

```
# lpadmin -p printer-name -f allow:form-list ↵
# lpadmin -p printer-name -f deny:form-list ↵
```

The *form-list* is a comma or space separated list of names of forms. If you use spaces to separate names, enclose the entire list (including the **allow:** or **deny:** but not the **-f**) in quotes.

The first command shown above adds names to the allow list and removes them from the deny list. The second command adds names to the deny list and removes them from the allow list. To make the use of all forms permissible, specify **allow:all**; to deny permission for all forms, specify **deny:all**.

If you do not use this option, the LP print service will consider that the printer denies the use of all forms. It will, however, allow you to mount any form, thereby making it implicitly available to use. (See “Mounting a form or print wheel” later in this chapter for more information.)

Printer fault alerting

This section does not apply if you are making a remote printer accessible to users on your system. The LP print service provides a framework for detecting printer faults and alerting you to them. Faults can range from simple problems, such as running out of paper or ribbon, or needing to replace the toner, to more serious faults, such as a local power failure or a printer failure. The range of fault indicators is also broad, ranging from dropping carrier (the signal that indicates that the printer is on line), to sending an XOFF, to sending a message. Only two classes of printer fault indicators are recognized by the LP print service itself: a drop in carrier and an XOFF not followed in reasonable time by an XON. However, you can add filters that recognize any other printer fault

indicators, and rely on the LP print service to alert you to a fault when the filter detects it. For a description of how to add a filter and how a filter should let the LP print service know a fault has occurred, see Chapter 9.

You can choose one of several ways to receive an alert to a printer fault:

- You can receive an alert via electronic mail. See the manual page for the **mailx(1)** command.
- You can receive an alert written to any terminal on which you are logged in. See the manual page for the **write(1)** command.
- You can receive an alert through a program of your choice.
- You can receive no alerts.

If you elect to receive no alerts, you will need a way of finding out about the faults and fixing them; the LP print service will not continue to use a printer that has a fault. In addition to the method of alerting, you can also arrange for repeated alerts every few minutes until the fault is cleared. You can choose the rate of repeated alerts, or you can choose to receive only one alert per fault. Without a filter that provides better fault detection, the LP print service cannot automatically determine when a fault has been cleared except by trying to print another file. It will assume that a fault has been cleared when it is successfully able to print a file. Until that time, if you have asked for only one alert per fault, you will not receive another alert. If, after you have fixed a fault, but before the LP print service has tried printing another file, the printer faults again, or if your attempt to fix the fault fails, you will not be notified. Receiving repeated alerts per fault, or requiring manual re-enabling of the printer (see “Printer fault recovery,” below), will overcome this problem.

To arrange for alerting to a printer fault, enter one of the following commands:

```
# lpadmin -p printer-name -A mail -W minutes ↵  
# lpadmin -p printer-name -A write -W minutes ↵  
# lpadmin -p printer-name -A'command' -W minutes ↵
```

The first two commands direct the LP print service to send you a mail message or write the message directly to your terminal, respectively, for each alert. The third command directs the LP print service to run the *command* for each alert. The shell environment currently in effect when you enter the third command is saved and restored for the execution of *command*. The environment includes environment variables, user and group IDs, and current directory. The *minutes* argument is the number of minutes between repeated alerts.

If you do not want the LP print service to issue an alert when a fault occurs, enter the following:

```
# lpadmin -p printer-name -A none ↵
```

If you want mail sent or a message written to another user when a printer fault occurs, use the third command with the option **-A'mail username'** or **-A'write username'**.

Once a fault occurs and you start receiving repeated alerts, you can direct the LP print service to stop sending you alerts (for the current fault only), by executing the following command:

```
# lpadmin -p printer-name -A quiet ↵
```

If the *printer-name* is **all** in any of the commands above, the alerting condition will apply to all printers.

If you don't define an alert method, you will receive mail once for each printer fault. If you define a method without the **-W** option, you will be alerted once for each fault.

Printer fault recovery

This section does not apply if you are making a remote printer accessible to users on your system. When a printer fault has been fixed and the printer is ready for printing again or when the fault is unrecoverable, the LP print service will respond in one of four ways:

- Continue where printing stopped.
- Restart printing at the beginning of the print request that was active when the fault occurred.
- Wait for you to tell the LP print service to re-enable the printer.
- Abort the job and not re-schedule it.

To specify the way the LP print service should recover after a fault has been cleared or if the fault is unrecoverable, enter one of the following commands:

```
# lpadmin -p printer-name -F continue ↵
# lpadmin -p printer-name -F beginning ↵
# lpadmin -p printer-name -F wait ↵
# lpadmin -p printer-name -F abort ↵
```

These commands direct the LP print service, respectively, to continue at the top of the page, restart from the beginning, wait for you to enter an **enable** command to re-enable the printer (see

“Enabling and disabling printer” for information on the **enable** command), or abort the job.

If you do not specify how the LP print service is to resume after a printer fault is cleared, it will try to continue where printing stopped, or, failing that, at the beginning of the print request.

If the recovery is **continue**, but the interface program does not stay running so that it can detect when the printer fault has been cleared, printing will be attempted every few minutes until it succeeds. You can force the LP print service to retry immediately by issuing an **enable** command.

User access restrictions

You can control which users are allowed to use a particular printer on your system. For instance, if you're designating one printer to handle sensitive information, you don't want all users to be able to use the printer. Another time you might want to do this is when you're authorizing the use of a high quality printer which produces expensive output.

The LP print service will use a list of users allowed or denied access to a printer. The LP print service will reject a user's request to print a file on a printer he or she is not allowed to use. If your users have access to remote printers, or if users on other systems have access to printers on your system, make sure that the allow and deny lists for those printers on your computer match the allow and deny lists on the remote system where the remote printers reside. If these two sets of lists don't match, your users may receive conflicting messages (some accepting jobs, and others refusing jobs) when submitting requests to remote printers.

The method of listing the users allowed or denied access to a printer is similar to the method used to list users allowed or denied access to the **cron** and **at** facilities, and the method described above in “Forms Allowed.” Briefly, the rules are as follows:

- An allow list is a list of users allowed to use the printer. A deny list is a list of users denied access to the printer.
- If the allow list is not empty, only the users listed are allowed; the deny list is ignored. If the allow list is empty, users listed in the deny list are not allowed. If both lists are empty, there are no restrictions on who may use the printer.
- Specifying **all** in the allow list allows everybody access to the printer; specifying **all** in the deny list denies access to everybody except the user **lp** and the super-user (**root** or **sysadm**).

You can add names of users to either list using one of the following commands:

```
# lpadmin -p printer-name -u allow:user-list ↵
# lpadmin -p printer-name -u deny:user-list ↵
```

The *user-list* is a comma or space separated list of names of users. If you use spaces to separate the names, enclose the entire list (including the **allow:** or **deny:** but not the **-u**) in quotes. Each item in the *user-list* may take any of the following forms:

<i>user</i>	<i>user</i> on any system
all	All users on all systems
<i>systemuser</i>	<i>user</i> on <i>system</i> only
<i>user</i>	<i>user</i> on local system only
alluser	<i>user</i> on any system
all!all	All users on all systems
system!all	All users on <i>system</i>
!all	All users on local system

The first command shown above adds the names to the allow list and removes them from the deny list. The second command adds the names to the deny list and removes them from the allow list.

If you do not use this option, the LP print service will assume that everybody may use the printer.

Banner pages

Most users want to have the output of each print request preceded by a banner page. A banner page shows who requested the printing, the request ID for it, and when the output was printed. It also allows for an optional title that the requester can use to better identify a printout. Finally, the banner page greatly eases the task of separating a sequence of print requests so that each may be given to the correct user.

Sometimes a user needs to avoid printing a banner page. The likely occasions are when the printer has forms mounted that should not be wasted, such as payroll checks or accounts payable checks. Printing a banner page under such circumstances may cause problems.

Enter the following command to allow users to request no banner page with the **lp -o nobanner** command:

```
# lpadmin -p printer-name -o nobanner ↵
```

If you later change your mind, you can reverse this choice by entering the following command:

```
# lpadmin -p printer-name -o banner ↵
```

If you do not allow a user to skip the banner page, the LP print service will reject all attempts to avoid a banner page when printing on the printer. This is the default action.

If you want to disable banner page printing for all print requests on a specific printer, enter the following command:

```
# lpadmin -p printer-name -o banneroff ↵
```

Printer description

An easy way to give users of the LP print service helpful information about a printer is by adding a description of it. This description can contain any message you like, including the number of the room where the printer is found, the name of the person to call with printer problems, and so forth.

Users can see the message when they use the **lpstat -D -p printer-name** command.

To add a description of a printer, enter the following command.

```
# lpadmin -p printer-name -D 'text' ↵
```

The *text* is the message. You'll need to include the quotes if the message contains blanks or other characters that the shell might interpret if the quotes are left out.

Default printing attributes

The attributes of a printing job include the page size, print spacing (character pitch and line pitch), and **stty** options for that job. If a user requests a job to be printed on a particular form, the printing attributes defined for that form will be used for that job. When, however, a user submits a print request without requesting a form, the print service uses one of several sets of default attributes.

- If the user has specified attributes to be used, those attributes will take precedence.
- If the user has not specified attributes, but the administrator has executed the **lpadmin -o** command, the default attributes for that command will take precedence.
- If neither of the above, the attributes defined in the **terminfo** database for the printer being used will take precedence.

The LP print service lets you override the defaults for each printer. Doing so can make it easier to submit print requests by allowing

you to designate different printers as having different default page sizes or print spacing. A user can then simply route a file to the appropriate printer to get a desired style of output. For example, you can have one printer dedicated to printing wide (132-column) output, another printing normal (80-column by 66-line) output, and yet another printing letter quality (12 characters per inch, 8 lines per inch) output.

You can independently specify four default settings, page width, page length, character pitch, and line pitch. You can scale these to fit your needs: the first two can be given in characters and lines, or inches or centimeters. The last two can be given as characters and lines per inch or per centimeter. In addition, the character pitch can be specified as **pica** for 10 characters per inch (cpi), **elite** for 12 cpi, or **compressed** for the maximum cpi the printer can provide (up to a limit of 30 cpi).

Set the defaults using one or more of the following commands:

```
# lpadmin -p printer-name -o width=scaled-number ↵
# lpadmin -p printer-name -o length=scaled-number ↵
# lpadmin -p printer-name -o cpi=scaled-number ↵
# lpadmin -p printer-name -o lpi=scaled-number ↵
```

Append the letter **i** to the *scaled-number* to indicate inches, or the letter **c** to indicate centimeters. The letter **i** for character pitch (**cpi**) or line pitch (**lpi**) is redundant. You can also give **pica**, **elite**, or **compressed** instead of a number for the character pitch.

If you don't provide defaults when you configure a printer, then the page size and print spacing will be taken from the data for your printer type in the **terminfo** database. (If you do not specify a printer type, the type will be **unknown**, for which there is an entry in the **terminfo** database.) You can find out what the defaults will be by first defining the printer configuration without providing your own defaults, then using the **lpstat** command to display the printer configuration. The following command reports the default page size and print spacing:

```
# lpstat -p printer-name -l ↵
```

Printer class membership

This section does not apply if you are making a remote printer accessible to users on your system. It is occasionally convenient to treat a collection of printers as a single class. The benefit is that a user can submit a file for printing by a member of a class, and the LP print service will pick the first printer in the class that it finds

free. This allows faster turn-around, as printers are kept as busy as possible.

Classes aren't needed if the only purpose is to allow a user to submit a print request by type of printer. The `lp -T content-type` command allows a user to submit a file and specify its type. The first available printer that can handle the type of the file will be used to print it. The LP print service will avoid using a filter, if possible, by choosing a printer that can print the file directly over one that would need it filtered first. See Chapter 9 for more information about filters.

Classes do have uses, however. One use is to put into a class a series of printers that should be used in a particular order. If you have a high speed printer and a low speed printer, for instance, you probably want the high speed printer to handle as many print requests as possible, with the low speed printer reserved for use when the other is busy. Because the LP print service always checks for an available printer in the order the printers were added to a class, you could add the high speed printer to the class before the low speed printer, and let the LP print service route print requests in the order you wanted.

Until you add a printer to a class, it doesn't belong to one. If you want to do so, use the following command:

```
# lpadmin -p printer-name -c class-name ↵
```

If the class *class-name* doesn't exist yet, it will be created. If you want to remove a printer from a class without deleting the printer, enter the following command:

```
# lpadmin -p printer-name -r class-name ↵
```

The class name may contain a maximum of fourteen alphanumeric characters and underscores. Class names and printer names must be unique. Because they are, a user can specify the destination for a print request without having to know whether it's a class of printers or a single printer.

System default destination

You can define the printer or class to be used to print a file when the user has not explicitly asked for a particular destination and has not set the `LPDEST` shell variable. The printer or class must already exist.

Make a printer or class the default destination by entering the following command:

```
# lpadm -d printer-or-class-name ↵
```

If you later decide that there should be no default destination, enter a null *printer-or-class-name* as in the following command:

```
# lpadm -d ↵
```

If you don't set a default destination, there will be none. Users will have to explicitly name a printer or class in each print request, (unless they specify the **-T content-type** option) or will have to set the LPDEST shell variable with the name of a destination.

Mounting a form or print wheel

See Chapter 9 for information about pre-printed forms. Before the LP print service can start printing files that need a preprinted form or print wheel, you must physically mount the form or print wheel on a printer, and notify the LP print service that you have mounted it. (It is not necessary for a form to be included on the allow list in order to mount it.) If alerting has been set on the form or print wheel, you will be alerted when enough print requests are queued waiting for it to be mounted. (See “Alerting to mount a form” below and “Alerting to mount a print wheel” above.)

When you mount a form you may want to see if it is lined up properly. If an alignment pattern has been defined for the form, you can ask that this be repeatedly printed after you've mounted the form, until you have adjusted the printer so that the alignment is correct.

Mounting a form or print wheel involves first loading it onto the printer and then telling the LP print service that it is mounted. Because it is difficult to do this on a printer that's currently printing, and because the LP print service will continue to print files not needing the form on the printer, you will probably have to disable the printer first. The proper procedure is as follows:

1. Disable the printer, using the **disable** command.
2. Mount the new form or print wheel as described immediately after this list.
3. Re-enable the printer, using the **enable** command. (The **disable** and **enable** commands are described in “Enabling and Disabling a Printer.”)

First, physically load the new form or print wheel into the printer. Then enter the following command to tell the LP print service it has been mounted.

Enter the following command if you are mounting a form:

```
# lpadmin -p printer-name -M -f form-name -a -o filebreak ↵
```

Enter the following command if you are mounting a print wheel:

```
# lpadmin -p printer-name -M -S print-wheel-name ↵
```

If you are mounting a form with an alignment pattern defined for it, you will be asked to press the Enter key before each copy of the alignment pattern is printed. After the pattern is printed, you can adjust the printer and press the Enter key again. If no alignment pattern has been defined, you won't be asked to press the Enter key. You can drop the **-a** and **-o filebreak** options if you don't want to bother with the alignment pattern.

The **-o filebreak** option tells the LP print service to add a form-feed after each copy of the alignment pattern. The actual control sequence used for the form-feed depends on the printer involved and is obtained from the **terminfo** database. If the alignment pattern already includes a form-feed, leave out the **-o filebreak** option.

To unmount a form, use the following command:

```
# lpadmin -p printer-name -M -f none ↵
```

To unmount a print wheel, use the following command:

```
# lpadmin -p printer-name -M -S none ↵
```

Until you've mounted a form on a printer, only print requests that don't require a form will be printed. Likewise, until you've mounted a print wheel on a printer, only print requests that don't require a particular print wheel will be printed. Print requests that do require a particular form or print wheel will be held in a queue until the form or print wheel is mounted.

Removing a printer or class

You can remove a printer or class if it has no pending print requests. If there are pending requests, you have to first move them to another printer or class (using the **lpmove** command), or cancel them (using the **cancel** command).

Removing the last remaining printer of a class automatically removes the class as well. The removal of a class, however, does not cause the removal of printers that were members of the class. If the printer or class removed is also the system default destination, the system will no longer have a default destination.

To remove a printer or class, enter the following command:

```
# ladmin -x printer-or-class-name ↵
```

If all you want to do is to remove a printer from a class without deleting that printer, enter the following command:

```
# ladmin -p printer-name -r class-name ↵
```

Putting it all together

You can add a new printer by completing a number of separate steps, shown in the commands described above. You may find it easier, however, to enter one or two commands that combine all the necessary arguments. Below are some examples.

Example 1

Add a new printer called **lp1** (of the type **455**) on printer port **/dev/tty13**. It should use the standard interface program, with the default page size of 90 columns by 71 lines, and linefeeds should *not* be mapped into carriage return/linefeed pairs.

```
# ladmin -p lp1 -v /dev/tty13 -T 455 -o "width=90 length=71 stty=-onlcr" ↵
```

Example 2

Add a new printer called **laser** on printer port **/dev/tty41**. It should use a customized interface program, located in the directory **/usr/doceng/laser_interface**. It can handle three file types—**i10**, **i300**, and **impress**—and it may be used only by the users **doceng** and **docpub**. (The following command line is broken over multiple lines for readability.)

```
# ladmin -p laser -v /dev/tty41 -i /usr/doceng/laser_interface \  
-I "i10,i300,impress" -u "allow:doceng,docpub" ↵
```

Example 3

When you added the **lp1** printer in the first example, you did not set the alerting. Do this now: have the LP print service alert you—by writing to the terminal on which you are logged in—every 10 minutes after a fault until you fix the problem.

```
# ladmin -p lp1 -A write -W 10 ↵
```

Examining a printer configuration

Once you've defined a printer configuration, you probably want to review it to see if it is correct. If after examining the configuration you find you've made a mistake, just reenter the command that applies to the part that's wrong.

Use the **lpstat** command to examine both the configuration and the current status of a printer. The short form of this command gives just the status; you can use it to see if the printer exists and if it is busy, idle, or disabled. The long form of the command gives a complete configuration listing. See Chapter 9 for how to get MAC information about print requests and printers on secure systems.

Enter one of the following commands to examine a printer.

```
# lpstat -p printer-name ↵
# lpstat -p printer-name -l ↵
```

(The second command is the long form.) With either command you should see one of the following lines of output.

```
printer printer-name now printing request-id.enabled since date.
```

```
printer printer-name is idle. enabled since date.
```

```
printer printer-name disabled since date.
reason
```

```
printer printer-name waiting for auto-retry.
reason
```

The message waiting for auto-retry shows that the LP print service failed in trying to use the printer (because of the *reason* shown), and that it will try again later.

With the long form of the command, you should also see the following output:

```

Form mounted: form-name
Content types: content-type-list
Printer type: printer-type
Description: comment
Connection: connection-info
Interface: pathname
On fault: alert-method
After fault: fault-recovery
Users allowed:
    user-list
Forms allowed:
    form-list
Banner required
Character sets:
    character-set-list
Default pitch: integer CPI, integer LPI
Default page size: scaled-decimal wide, scaled-decimal long
Default port settings: stty-option-list

```

Making printers available

There are two steps in making a printer ready for use after you've defined the printer configuration. First, you must instruct the LP print service to accept print requests for the new printer. To do this, run the **accept** command. Second, you must activate or enable the new printer. To do this, run the **enable** command. These tasks are separate steps because you may need to do one but not the other.

Accepting print requests for a new printer

Telling the LP print service to accept print requests for the new printer is done with the **accept** command. For more information on this command, see Chapter 9. For now, all you need to know is that you should enter the following command to let this printer be used.

```
# accept printer-or-class-name ↵
```

As you can see, this command is needed to let the LP print service start accepting print requests for a class, too. To prevent the print service from accepting any more requests, execute the following command.

```
# reject printer-or-class-name ↵
```

Enabling and disabling a printer

Because you may want to make sure, before printing begins, that the correct form is loaded in your printer, the correct print wheel or

font cartridge is in place, and the printer is on-line, the LP print service will wait for an explicit signal from you before it starts printing files. Once you have verified that all the necessary components are in place, you can request the beginning of printing by issuing the **enable** command for a particular printer, as follows:

```
# enable printer-name ↵
```

If you want to enable several printers simultaneously, list the printers (separating the names with spaces) on the same line as the **enable** command. Don't enclose the list in quotes.

Disabling a printer stops further print requests from being printed. (It does not, however, stop the LP print service from accepting new print requests for the printer.) From time to time you may want to disable a printer. For example, you may want to interrupt a print request, or you may want to change a form or print wheel, in which case you should disable the printer first. Normally, disabling a printer also stops the request that's currently being printed, placing it back in the queue so it can be printed later. You can, however, have the LP print service wait until the current request finishes, or even cancel the request outright.

To disable a printer, enter one of the following commands:

```
# disable -r "reason" printer-name ↵  
# disable -W -r "reason" printer-name ↵  
# disable -c -r "reason" printer-name ↵
```

The first command disables the printer, stopping the currently printing request and saving it for printing later. The other commands also disable the printer, but the second one makes the LP print service wait for the current request to finish, while the third cancels the current request. The **-c** and **-W** options are not valid when the **disable** command is run to stop a remote printer because, when run for a remote printer, **disable** stops the transferring (rather than the actual printing) of print requests. The *reason* is stored and displayed whenever anyone checks the status of the printer. You can omit it (and the **-r** option) if you don't want to specify a reason.

Several printers can be disabled at once by listing their names in the same line as the **disable** command. You can only enable or disable local printers; the loading of forms, print wheels, and cartridges in a remote printer and the enabling of that printer are the responsibility of the administrator of the remote system. You can, however, enable or disable the transfer of print requests to the remote system on which a printer is located. Only individual printers can be enabled and disabled; classes cannot.

Allowing users to enable and disable a printer

You may want to make the **enable** and **disable** commands available for use by other users. This availability is useful, for instance, if you have a small organization where anyone who spots a problem with the printer should be able to disable it and fix the problem. This is *not* a good idea if you want to keep others from interfering with the proper operation of the print services.

If you want to allow others access to the **enable** and **disable** commands, use a standard DG/UX system feature called the *setuid bit*. By assigning ownership of these commands to the user **lp** (this should have been done automatically when you installed the software), and by setting the setuid bit, you can make sure that anyone will be allowed to use the **enable** and **disable** commands. Clearing the bit removes this privilege.

To allow everybody to run **enable** and **disable**, enter the following two commands:

```
# chown lp /usr/bin/enable /usr/bin/disable ↵  
# chmod u+s /usr/bin/enable /usr/bin/disable ↵
```

The first command makes the user **lp** the owner of the commands; this step should be redundant, but it is safer to run the command than to skip it. The second command turns on the setuid bit.

To prevent others from running **enable** and **disable**, enter the following command:

```
# chmod u-s /usr/bin/enable /usr/bin/disable ↵
```

End of Chapter

9

Administering the LP subsystem

This chapter describes how to maintain print services with the LP print service utilities. It also discusses how to handle PostScript printers. The chapter includes the following sections:

- Providing forms
- Providing filters
- Managing the printing load
- Starting and stopping the LP print service
- Managing the LP print service logs
- Managing PostScript printers
- Customizing the print service
- Managing the line printer subsystem on a trusted DG/UX system

Table 9–1 is a summary of the commands required to maintain print services with LP. These commands are located in the `/usr/bin` directory unless otherwise noted. You must be the superuser (`sysadm` or `root`) to use these commands.

Table 9–1 LP print service command reference

Action	Command
Activating a printer:	<code>enable(1)</code>
Canceling a request for a file to be printed:	<code>cancel(1)</code>
Sending a file (or files) to a printer:	<code>lp(1)</code>
Reporting the status of the LP print service:	<code>lpstat(1)</code>
Deactivating specified printers:	<code>disable(1)</code>
Permitting job requests to be queued for a specific destination:	<code>accept(1M)</code>
Preventing jobs from being queued for a specified destination:	<code>reject(1M)</code>
Setting up or changing printer configurations:	<code>lpadmin(1M)</code>
Setting up or changing filter definitions:	<code>lpfilter(1M)</code>
Setting up or changing preprinted forms:	<code>lpforms(1M)</code>
Mounting a form:	<code>lpadmin(1M)</code>

Continued

Table 9-1 LP print service command reference

Action	Command
Moving output requests from one destination to another:	lpmove(1M)
Starting the LP print service scheduler:	lpsched(1M)
Stop the LP print service scheduler:	lpshut(1M)
Setting or changing the default priority and priority limits that can be requested by users of the LP print service:	lpusers(1M)

Providing forms

A form is a sheet of paper, on which text or graphical displays have already been printed, that can be loaded into a local printer (that is, a printer on your system) for use in place of plain stock. Common examples of forms include company letterhead, special paper stock, invoices, blank checks, vouchers, receipts, and labels.

Typically, several copies of a blank form are loaded into a printer, either as a tray of single sheets or as a box of fan-folded paper. An application is used to generate data that will be printed on the form, thereby filling it out.

The LP print service helps you manage the use of preprinted forms, but does not provide your application any help in filling out a form; this is solely your application's responsibility. The LP print service, however, will keep track of which print requests need special forms mounted and which forms are currently mounted. It can alert you to the need to mount a new form.

This section tells you how you can manage the use of preprinted forms with the LP print service. You will see how you can

- define a new form
- change the print service's description of an existing form
- remove the print service's description of a form
- examine the print service's description of a form
- restrict user access to a form
- arrange alerting to the need to mount a form
- inform the print service that a form has been mounted

Defining a form

When you want to provide a new form, the first thing you have to do is define its characteristics. To do so, enter information about

each of the nine required characteristics (page length, page width, and so on) as input to the **lpforms** command (see below for details). The LP print service will use this information for two purposes: to initialize the printer so that printing is done properly on the form, and to send you reminders about how to handle that form. Before running the **lpforms** command, gather the following information about your new form:

Page length	The length of the form, or of each page in a multi-page form. This can be expressed as the number of lines, or the size in inches or centimeters.
Page width	The width of the form, expressed in characters, inches, or centimeters.
Number of pages	The number of pages in a multi-page form. The LP print service uses this number with a filter (if available) to restrict the alignment pattern to a length of one form. (See the description of alignment patterns below.) If no filter is available, the LP print service does not truncate the output.
Line pitch	A measurement that shows how closely together separate lines appear on the form. It can be expressed in either lines per inch or lines per centimeter.
Character pitch	A measurement that shows how closely together separate characters appear on the form. It can be expressed in either characters per inch or characters per centimeter.
Character set choice	The character set, print wheel, or font cartridge that should be used when this form is used. A user may choose a different character set for his or her own print request when using this form, or you can insist that only one character set be used.
Ribbon color	If the form should always be printed using a certain color ribbon, then the LP print service can remind you which color to use when you mount the form.
Comment	Any comment you wish to make about the form. This comment is available for users to see so they can understand what the form is, when it should be used, and so on.

Alignment pattern A sample file that the LP print service uses to fill one blank form. When mounting the form, you can print this pattern on the form to align it properly. You can also define a content type for this pattern so that the printer service knows how to print it.

The LP print service does not try to mask sensitive information in an alignment pattern. If you do not want sensitive information printed on sample forms (very likely the case when you align checks, for instance) then you should mask the appropriate data. The LP print service keeps the alignment pattern stored in a safe place, where only you (that is, the user **lp** and the super-user **root** or **sysadm**) can read it.

When you've gathered this information about the form, enter it as input to the **lpforms** command. You may want to record this information first in a separate file so you can edit it before entering it with **lpforms**. You can then use the file as input instead of typing each piece of information separately after a prompt. Whichever method you use, enter the information in the following format:

Page length: *scaled-number*
Page width: *scaled-number*
Number of pages: *integer*
Line pitch: *scaled-number*
Character pitch: *scaled-number*
Character set choice: *character-set-name*[**,mandatory**]
Ribbon color: *ribbon-color*
Comment:
comment
Alignment pattern: [*content-type*]
alignment-pattern

Although these attributes are described in detail on the previous page, a few points should be emphasized here. First, the phrase **mandatory** is optional and, if present, means that the user cannot override the character set choice in the form.

Second, **content-type** can be given optionally, with an alignment pattern. If this attribute is given, the print service uses the alignment pattern specified to determine, as necessary, how to filter and print the file.

With two exceptions, the information in the above list may appear in any order. The exceptions are the alignment pattern (which must always appear last) and *comment* (which must always follow the line with the **Comment:** prompt). If the *comment* contains a line beginning with a key phrase (such as **Page length**, **Page width**, and so on), precede that line with a > character so the key phrase is

hidden. Be aware, though, that any initial > will be stripped from the comment when it is displayed.

Not all of the information has to be given. Table 9–2 shows the defaults for the form’s various attributes.

Table 9–2 Default printer form attributes

Attribute	Default
Page length	66 lines
Page width	80 columns
Number of pages	1
Line pitch	6 per inch
Character pitch	10 per inch
Character set choice	any
Ribbon color	any
Comment	(no default)
Alignment pattern	(no default)

To define the form, use one of the following commands

```
# lpforms -f form-name -F file-name ↵
# lpforms -f form-name - ↵
```

where *file-name* is the full path for the file.

The first command gets the form definition from a file; the second command gets the form definition from you, through the standard input. A *form-name* can be anything you choose, as long as it contains a maximum of fourteen alphanumeric characters and underscores.

If you need to change a form, just reenter one of the above commands. You need only provide information for items that must be changed; items for which you don’t specify new information will stay the same.

Removing a Form

The LP print service imposes no fixed limit on the number of forms you may define. It is a good idea, however, to remove forms that are no longer appropriate. If you don’t, users will see a long list of obsolete forms when choosing a form, and may be confused. In addition, because the LP print service must occasionally look

through all the forms listed before performing certain tasks, the failure to remove obsolete forms may require extra, unnecessary processing by the print service.

To remove a form, enter the following command:

```
# lpforms -f form-name -x ↵
```

Restricting user access

If your system has a form that you don't want to make available to everyone, you can limit its availability to selected users. For example, you may want to limit access to checks to the people in the payroll department or accounts payable department.

The LP print service restricts the availability of a form by using the list of users allowed or denied access to that form. If a user is not allowed to use a particular form, the LP print service will reject his or her request to print a file with it.

The method used to allow or deny users access to a form is similar to the method used to allow or deny users access to the **cron** and **at** facilities. See the manual pages for **crontab(1)** and **cron(1M)**. Briefly, the rules are as follows:

- An allow list shows the users who are allowed to use the form. A deny list shows users who are not allowed to use the form.
- If the allow list is not empty, only the users listed are allowed; the deny list is ignored. If the allow list is empty, the users listed in the deny list are not allowed to use the form. If both lists are empty, there are no restrictions on who may use the form.
- Specifying **all** in the allow list allows everybody to use the form; specifying **all** in the deny list allows no one except the user **lp** and the super-user (**root** or **sysadm**) to use the form.

If users on your system are to be able to access forms on a remote printer, it's necessary for all the users included on the allow list for the local system to be included on the allow list for the remote system, as well.

If, on the other hand, a local user is to be denied permission to use forms on a remote printer, it's not necessary for the deny lists on both the local and remote print services to include that user. By being included in only one of these deny lists, a user can be denied access to remote forms. As a courtesy to your users, however, it's a good idea to make sure that any local users who are included in a deny list on a remote system are included in the corresponding deny list on your local system. By doing this you can make sure that

whenever a user on your system requests a form that he or she is not authorized to use, he or she is immediately informed that permission to use the form is being denied. If the local print service does not “know” that a user is denied permission to use a particular remote form, there will be a delay before the user receives a “permission denied” message from the remote system.

You can add names of users to either list using one of the following commands:

```
# lpforms -f form-name -u allow:user-list ↵
# lpforms -f form-name -u deny:user-list ↵
```

The *user-list* is a comma or space separated list of names of users. If you use spaces to separate the names, enclose the entire list (including the **allow:** or **deny:** but not the **-u**) in quotes. Each item in the list can include a system name, as shown under “Restricting User Access” earlier in this chapter. The first command adds the names to the allow list and removes them from the deny list. The second command adds the names to the deny list and removes them from the allow list. Specifying **allow:all** will allow everybody; specifying **deny:all** will deny everybody.

If you do not add usernames to the allow or deny lists, the LP print service will assume that everybody may use the form.

Alerting to mount a form

If you define more forms than printers, you will obviously not be able to print files on all the forms simultaneously. This means that some print requests may be held in a queue until you mount the forms they need. How will you know when to mount a particular form? One method would be to periodically monitor the number of print requests pending for that form. The LP print service, however, provides an easier way: You can ask to be alerted when the number of requests waiting for a form has exceeded a specified threshold.

You can choose one of several ways to receive an alert.

- You can receive an alert via electronic mail. See the manual page for the **mailx(1)** command.
- You can receive an alert written to any terminal on which you are logged in. See the manual page for the **write** command.
- You can receive an alert through a program of your choice.
- You can receive no alerts.

If you elect to receive no alerts, you are responsible for checking to see if any print requests haven’t printed because the proper form

isn't mounted. In addition to the method of alerting, you can also set the number of requests that must be queued before you are alerted, and you can arrange for repeated alerts every few minutes until the form is mounted. You can choose the rate of repeated alerts, or choose to receive only one alert for each form.

To arrange for alerting to the need to mount a form, enter one of the following commands:

```
# lpforms -f form-name -A mail -Q requests -W minutes ↵
# lpforms -f form-name -A write -Q requests -W minutes ↵
# lpforms -f form-name -A 'command' -Q requests -W minutes ↵
```

The first two commands direct the LP print service to send you a mail message or write the message directly to your terminal, respectively, for each alert. The third command directs the LP print service to run the *command* for each alert. The shell environment in effect when you enter the third command is saved and restored for the execution of *command*; this includes the environment variables, user and group IDs, and the current directory.

In each command line, the argument *requests* is the number of requests that need to be waiting for the form to trigger the alert, and the argument *minutes* is the number of minutes between repeated alerts. If you want mail sent or a message written to another user when a printer alert occurs, use the third command with the option **-A'mail username'** or **-A'write username'**.

If you want the print service to issue no alert when the form needs to be mounted, execute the following command:

```
# lpforms -f form-name -A none ↵
```

When you start receiving repeated alerts, you can direct the LP print service to stop sending you alerts (for the current case only) by issuing the following command:

```
# lpforms -f form-name -A quiet ↵
```

Once the form has been mounted and unmounted again, alerts will resume if too many requests are waiting. Alerts will also start again if the number of requests waiting falls below the **-Q** threshold and then rises up to the **-Q** threshold again. This happens when waiting requests are canceled, and when the type of alerting is changed.

If *form-name* is **all** in any of the commands above, the alerting condition applies to all forms for which an alert has not already been defined.

If you don't define an alert method for a form, you will not receive an alert to mount it. If you define a method without the **-W** option, you will be alerted once for each occasion.

Mounting a form

For information on mounting a form, see Chapter 8.

Examining a form

Once you've defined a form to the LP print service, you can examine it with one of two commands, depending on the type of information you want to check. The **lpforms** command displays the attributes of the form. (The display produced by **lpforms** can be used as input; you may want to save it in a file for future reference.) The **lpstat** command displays the current status of the form.

Enter one of the following commands to examine a defined form.

```
# lpforms -f form-name -l ↵
# lpforms -f form-name -l > file-name ↵
# lpstat -f form-name ↵
# lpstat -f form-name -l ↵
```

The first two commands present the definition of the form; the second command captures this definition in a file, which can be used later to redefine the form if you inadvertently remove the form from the LP print service. The last two commands present the status of the form, with the second of the two giving a long form of output, similar to the output of **lpforms -l**:

```
Page length: scaled-number
Page width: scaled-number
Number of pages: integer
Line pitch: scaled-number
Character pitch: scaled-number
Character set choice: character-set[,mandatory]
Ribbon color: ribbon-color
Comment:
comment
Alignment pattern: [ content-type]
content
```

To protect potentially sensitive content, the alignment pattern is not shown if the **lpstat** command is used.

Providing filters

This section explains how you can manage the use of filters with the LP print service. You will see how you can

- define a new filter
- change a filter
- remove a filter
- examine a filter

“Customizing the Print Service,” at the end of this chapter, describes how you can write a filter. First, let’s see what a filter is and how the LP print service can use one.

What is a filter?

A filter is a program that you can use for any of three purposes:

- To convert a user’s file from one data format to another so that it can be printed properly on a given printer
- To handle the special modes of printing that users may request with the `-y` option to the `lp` command (such as two-sided printing, landscape printing, draft or letter quality printing)
- To detect printer faults and notify the LP print service of them, so that the print service can alert you

Not every filter can perform all three tasks. Given the printer-specific nature of these three roles, the LP print service has been designed so that these roles can be implemented separately. This separation allows you or a printer manufacturer (or another source) to provide filters without having to change the LP print service.

A default filter is provided with the LP print service to provide simple printer fault detection; it does not convert files or handle any of the special modes. It may, however, be adequate for your needs.

Let’s examine the three tasks performed by filters more closely.

Task 1: converting files

For each printer (local or remote) you can specify what file content types it can print. When a user submits a file to print on any printer, and specifies its content type, the print service will find a printer that can handle files of that content type. Because many applications can generate files for various printers, this is often sufficient. However, some applications may not generate files that can be printed on your printers.

By defining and creating a filter that converts such files into a type that your printers can handle, you can begin to support more applications in the LP print service. (The LP print service comes

with a few filters for converting various types of files into PostScript.) For each filter you add to the system, you must specify one or more types of input it can accept and the type of output it can produce (usually only one).

When a user specifies (by executing `lp -T`) a file content type that no printer can handle, the print service tries to find a filter that can convert the file into an acceptable type. If the file to be printed is passed through a filter, the print service will then match the output type of that filter with a printer type or the input type of another filter. The LP print service will continue to match output types to input types in this way, thus passing a file through a series of filters, until the file reaches a printer that accepts it.

Below are some examples.

Example 1 —

The user Chris has run a spreadsheet program and has generated a file containing a copy of a spreadsheet. Chris now wants to print this file using the LP print service. You have only HP LaserJet printers on your system. Fortunately, the spreadsheet application understands how to generate output for several printers, and Chris knows it's necessary to request output that can be handled by the HP LaserJet. When Chris submits the file for printing, the LP print service queues it for one of the printers; no filter is needed.

Example 2 —

Marty has created a graphic image that can be displayed on a Tektronix 4014 terminal. Marty now wants to print this image, but all of the printers are PostScript printers. Fortunately, your system provides a filter called `posttek` that converts **Tektronix** type files to **PostScript**. Because you set the printer type to **PostScript**, the LP print service recognizes that it can use the `posttek` filter to convert Marty's output before printing it.

Task 2: handling special modes

Another important role that filters can perform is the handling of special printing modes. Each filter you add to the filter table can be registered to handle special modes and other aspects of printing:

- Special modes
- Printer type
- Character pitch
- Line pitch
- Page length
- Page width
- Pages to print

Character set
Form name
Number of copies

A filter is required to handle the special modes and printing of specific pages; the LP print service provides a default handling for all the rest. However, it may be more efficient to have a filter handle some of the rest, or it may be that a filter has to know several of these aspects to fulfill its other roles properly. A filter may need to know, for example, the page size and the print spacing if it is going to break up the pages in a file to fit on printed pages. As another example, some printers can handle multiple copies more efficiently than the LP print service, so a filter that can control the printer can use the information about the number of copies to skip the LP print service's default handling of multiple copies.

We'll see below how you can register special printing modes and other aspects of printing with each filter.

Task 3: detecting printer faults

Just as converting a file and handling special printing modes is a printer-specific role, so is the detecting of printer faults. The LP print service attempts to detect faults in general, and for most printers it can do so properly. The range of faults that the print service can detect by itself, however, is limited. It can check for hang-ups (loss of carrier, the signal that indicates the printer is on-line) and excessive delays in printing (receipt of an XOFF flow-control character to shut off the data flow, with no matching XON to turn the flow back on). However, the print service can't determine the cause of a fault, so it can't tell you what to look for.

A properly designed filter can provide better fault coverage. Some printers are able to send a message to the host describing the reason for a fault. Others indicate a fault by using signals other than the dropping of a carrier or the shutting off of data flow. A filter can serve you by detecting more faults and providing more information about them than you would otherwise receive.

Another service a filter can provide is to wait for a printer fault to clear and then to resume printing. This service allows for more efficient printing when a fault occurs because the print request that was interrupted does not have to be reprinted in its entirety. Only a real filter, which has knowledge of the control sequences used by a printer, can "know" where a file breaks into pages; thus only such a filter can find the place in the file where printing should resume.

The LP print service has a simple interface that allows a filter to send you fault information and to restart printing if it can. The alerting mechanism (see Chapter 8) is handled by the LP print

service; the interface program that manages the filter takes all error messages from the filter and places them in an alert message that can be sent to you. Thus you'll see any fault descriptions generated by the filter. If you've set the printer configuration so that printing should automatically resume after a fault is cleared, the interface program will keep the filter active, so that printing can pick up where it left off.

Will any program make a good filter?

It is tempting to use a program such as **troff**, **nroff**, or a similar word-processing program as a filter. However, the **troff** and **nroff** programs have a feature that allows references to be made in a source file to other files, known as *include files*. The LP print service does not recognize include files; it will not queue any that are referenced by a source file when that file is in a queue to be printed. As a result, the **troff** or **nroff** program, unable to access the include files, may fail. Other programs may have similar features that limit their use as filters.

Here are a few guidelines for evaluating a program for use as a filter:

- Only programs capable of reading data from standard input and writing data to standard output may be used as filters.
- Examine the kinds of files users will submit for printing that will require processing by the program. If they stand alone (that is, if they do not reference other files that the program will need), the program is probably okay.

Check also to see if the program expects any files other than those submitted by a user for printing. If it does, those files must be in the directory of the person using the filter, and they must be readable by all users authorized to use the filter. The latter prerequisite is necessary because filters are run with the user ID and group ID of the user who submitted the print request.

- If referenced files are permitted in the files submitted for printing, or if the program will need files other than those submitted by a user, then the program, unable to access the additional files, is likely to fail. We suggest you don't use the program under consideration as a filter; instead, have users run the program before submitting files for printing.

Referenced files that are always specified by full pathnames *may* be okay, but only if the filter is used for local print requests. When used on requests submitted from a remote system for printing on your system, the filter may still fail if the referenced files exist only on the remote system.

Filters for your system

The LP print service is delivered with several filters. As you add, change, or delete filters, you may overwrite or remove some of these original filters. If necessary, you can restore the original set of filters (and remove any filters you have added), with the following command:

```
# lpfilter -f all -i ↵
```

Defining a filter

When adding a new filter, the first thing you must do is to define the characteristics of its use. To do this, issue the **lpfilter** command with arguments that specify the values of the following filter characteristics:

- the name of the filter (that is, a command name)
- the types of input it will accept
- the types of output it will produce
- the types of printers to which it will be able to send jobs
- the names of specific printers to which it will send jobs
- the type of the filter (whether it's a **fast** filter or a **slow** filter)
- options

Each of these characteristics is described below.

Command —

This is the full path of the filter program. If there are any fixed options that the program always needs, include them here.

Input types —

This is the list of file content types that the filter can process. The LP print service doesn't impose a limit on the number of input types that can be accepted by a filter, but most filters can take only one. Several file types may be similar enough so that the filter can deal with them. You can use whatever names you like here, using a maximum of fourteen alphanumeric characters and dashes (not underscores). Because the LP print service uses these names to match a filter with a file type, you should follow a consistent naming convention. For example, if more than one filter can accept the same input type, use the same name for that input type when you specify it for each filter. These names should be advertised to your users so they know how to identify the type of a file when submitting that file for printing.

Output types —

This is the list of file types that the filter can produce as output. For each input type the filter will produce a single output type, of course; the output type may vary, however, from job to job. The names of the output types are also restricted to fourteen alphanumeric characters and dashes.

These names should either match the types of printers you have on your system, or match the input types handled by other filters. The LP print service groups filters together in a shell pipeline if it finds that several passes by different filters are needed to convert a file. It's unlikely that you will need this level of sophistication, but the LP print service allows it. Try to find a set of filters that takes (as input types) all the different files your users may want printed, and converts those files directly into types your printers can handle.

Printer types —

This is a list of printer types into which the filter can convert files. For most filters this list will be identical to the list of output types, but it can be different.

For example, you may have a printer that is given a single type for purposes of initialization (see Chapter 8), but which can recognize several different types of files. In essence this printer has an internal filter that converts the various types into one with which it can deal. Thus, a filter may produce one of several output types that match the "file types" that the printer can handle. The filter should be marked as working with that printer type.

As another example, you may have two different models of printers that are listed as accepting the same types of files. However, due to slight differences in manufacture, one printer deviates in the results it produces. You label the printers as being of different printer types, say A and B, where B is the one that deviates. You create a filter that adjusts files to account for the deviation produced by printers of type B. Because this filter is needed only for those printer types, you would list it as working only on type B printers.

For most printers and filters you can leave this part of the filter definition blank.

Printers —

You may have some printers that, although they're of the correct type for a filter, are in other ways not adequate for the output that the filter will produce. For instance, you may want to dedicate one printer for fast turn-around; only files that the printer can handle without filtering will be sent to that printer. Other printers, of

identical type, you allow to be used for files that may need extensive filtering before they can be printed. In this case, you would label the filter as working with only the latter group of printers.

In most cases a filter should be able to work with all printers that accept its output, so you can usually skip this part of the filter definition.

Filter type —

The LP print service recognizes **fast** filters and **slow** filters. Fast filters are labeled **fast** because they incur little overhead in preparing a file for printing, and because they must have access to the printer when they run. A filter that is to detect printer faults has to be a fast filter.

Slow filters are filters that incur a lot of overhead in preparing a file and that don't require access to a printer. The LP print service runs slow filters in the background, without tying up a printer. This allows files that don't need slow filtering to move ahead; printers will not be left idle while a slow filter works on a file if other files can be printed simultaneously.

Slow filters that are invoked by `modes` (via the `--y` option), must be run on the computer where the print request was issued. The LP print service can't pass values for `modes` to remote systems. It can, however, match a file content type (specified after the `-T` option of the `lp` command) to a content type on a remote system. Therefore, if you want to activate special modes on a remote system, you must do so by specifying content types that will allow the LP print service to match input types and output types.

Options —

Options specify how different types of information should be transformed into command line arguments to the filter command. This information may include specifications from a user (with the print request), the printer definition, and the specifications implemented by any filters used to process the request.

There are thirteen sources of information, each of which is represented by a *keyword*. Each option is defined in a *template*, a statement in the following format: *keyword pattern=replacement*. This type of statement is interpreted by the `lpfilter` command to mean "When the information referred to by *keyword* has the value matched by *pattern*, take the *replacement* string, expand any regular expressions it contains, and append the result to the command line."

The options specified in a filter definition may include none, all, or any subset of these thirteen keywords. In addition, a single

keyword may be defined more than once, if multiple definitions are required for a complete filter definition. (See “Defining options with templates” below.)

Supplying definitions to **lpfilter**

When you’ve gathered enough information to define the above characteristics of your filter, you are ready to run the **lpfilter** command, using your data as arguments. Because there are so many arguments, and because some of them may need to be entered more than once (with different values), we recommend you record this information first in a separate file and edit it, if necessary. You can then use the file as input to the **lpfilter** command and avoid typing each piece of information separately.

Whether you store the information in a file or enter it directly on the command line, use the following format:

Command: *command-pathname* [*options*]

Input types: *input-type-list*

Output types: *output-type-list*

Printer types: *printer-type-list*

Printers: *printer-list*

Filter type: **fast** or **slow**

Options: *template-list*

The information can appear in any order. Not all the information has to be given. When you do not specify values for the items appearing in Table 9–3, the values shown beside them are assigned by default.

Table 9–3 Filter defaults

Item	Default
Command:	(no default)
Input types:	any
Output types:	any
Printer types:	any
Printers:	any
Filter type:	slow
Options:	(no default)

As you can see, the default values define a very flexible filter, so you probably have to supply at least the input and output types. When

you enter a list, you can separate the items in it with blanks or commas, unless it is a *templates-list*; items in a *templates-list* must be separated by commas.

Defining options with templates

A template is a statement in a filter definition that defines an option to be passed to the filter command based on the value of one of the characteristics of the filter. A filter definition may include more than one template. Multiple templates may be entered on a single line and separated with commas, or they may be entered on separate lines, preceded by the **Options:** prefix.

The format of a template is as follows:

keyword pattern = replacement

The *keyword* identifies the type of option being registered for a particular characteristic of the filter.

Let's look at an example of how an option is defined for a particular filter. Suppose you want to have the print service scheduler assign print requests to filters on the basis of the following criteria:

- If the type of output to be produced by the filter is **impress**, then pass the **-I** option to the filter.
- If the type of output to be produced by the filter is **postscript**, then pass the **-P** option to the filter.

To specify these criteria, provide the following templates as options to the **lpfilter** command.

Options: OUTPUT impress=-I, OUTPUT postscript=-P

If the **Options:** line becomes too long, put each template on a separate line, as follows:

Options: OUTPUT impress=-I
Options: OUTPUT postscript=-P

In both templates, the *keyword* is defined as **OUTPUT**. In the first template, the value of *pattern* is **impress** and the value of the *replacement* is **-I**. In the second template, the value of *pattern* is **postscript** and the value of the *replacement* is **-P**.

Template keywords —

Table 9-4 shows the thirteen keywords available for defining options in a filter definition.

Table 9–4 Filter option keywords

Characteristic	Keyword	Possible patterns	Example
Content type (input)	INPUT	content–type	troff
Content type (output)	OUTPUT	content–type	postscript
Printer type	TERM	printer–type	att495
Printer name	PRINTER	printer–name	lp1
Character pitch	CPI	scaled–decimal	10
Line pitch	LPI	scaled–decimal	6
Page length	LENGTH	scaled–decimal	66
Page width	WIDTH	scaled–decimal	80
Pages to print	PAGES	page–list	1–5, 13–20
Character set	CHARSET	character–set	finnish
Form name	FORM	form–name	invoice2
Number of copies	COPIES	integer	3
Special modes	MODES	mode	landscape

To find out which values to supply for each type of template (that is, for the *pattern* and *replacement* arguments for each *keyword*), see the list below.

- The values for the INPUT and OUTPUT templates come from the file type that needs to be converted by the filter and the output type that has to be produced by the filter, respectively. They'll each be a type registered with the filter.
- The value for the TERM template is the printer type.
- The value for the PRINTER template is the name of the printer that will be used to print the final output.
- The values for the CPI, LPI, LENGTH, and WIDTH templates come from the user's request, the form being used, or the default values for the printer.
- The value for the PAGES template is a list of pages that should be printed. Typically, it is a comma separated list of page ranges, each of which consists of a dash separated pair of numbers or a single number (such as 1–5,6,8,10 for pages 1 through 5, 6, 8, and 10). However, whatever value was given in the **-P** option to a print request is passed unchanged.
- The value for the CHARSET template is the name of the character set to be used.

- The value for the FORM template is the name of the form requested by the `-f` option of the `lp` command.
- The value of the COPIES template is the number of copies that should be made of the file. If the filter uses this template, the LP print service will reduce to 1 the number of copies of the filtered file it will print, since this single copy will actually comprise the multiple copies produced by the filter.
- The value of the MODES template comes from the `-y` option of the `lp` command (the command used to submit a print request). Because a user can specify several `-y` options, there may be several values for the MODES template. The values will be applied in the left-to-right order given by the user.

The *replacement* part of a template shows how the value of a template should be given to the filter program. It is typically a literal option, sometimes with the place-holder `*` included to show where the value goes. The *pattern* and *replacement* can also use the regular expression syntax of `ed(1)` for more complex conversion of user input options into filter options. All of the regular expression syntax of `ed(1)` is supported, including the `\(... \)` and `\n` constructions, which can be used to extract portions of the *pattern* for copying into the *replacement*, and the `&`, which can be used to copy the entire *pattern* into the *replacement*. If a comma or an equals sign (`=`) is included in a *pattern* or a *replacement*, escape its special meaning by preceding it with a backslash (`\`). Note that some regular expressions include commas that will have to be escaped this way. A backslash in front of any of these characters is removed when the *pattern* or *replacement* is used.

The following examples show how this works.

Example 1 —

You provide the following filter definition for a filter called `col`.

Input types: N37, Nlp, simple

Output types: simple

Command: /usr/bin/col

Options: TERM 450 = -b, MODES expand = -x

Options: INPUT simple = -p -f

If you provide more than one definition (that is, more than one line) for any filter characteristic other than **OPTIONS**, only the second definition will be used by the print service.

After you have registered this definition with the print service by entering it as input with the `lpfilter` command, users' print requests will be handled as follows:

If a user enters the command

```
# lp -y expand report.dec10 ↵
```

the filter command will be run with the following arguments:

```
/usr/bin/col -x -p -f
```

If a user enters the command

```
# lp -T N37 -y expand report.dec10 ↵
```

the filter command will be run with the following arguments:

```
/usr/bin/col -x
```

Qualifier: The default printer is not of type 450.

If a user enters the command

```
# lp -y expand -T 450 report.dec10 ↵
```

the filter command will be run with the following arguments:

```
/usr/bin/col -b -x
```

Example 2 —

The filter program is called `/usr/lib/lp/postscript/dpost`. It takes one input type, **troff**, produces an output type called **postscript**, and works with any printer of type **postscript**. You've decided that your users need give just the abbreviations **port** and **land** when they ask for the paper orientation to be portrait mode and landscape mode, respectively. Because these options are not intrinsic to the LP print service, users must specify them using the **-y** option to the **lp** command.

The filter definition would look like this:

```
Input types: troff
Output types: postscript
Printer types: postscript
Filter type: slow
Command: /usr/lib/lp/postscript/dpost
Options: LENGTH * = -l *, CHARSET * = -s *
Options: MODES port = -o portrait, MODES land = -o
landscape
```

A user submitting a file of type **troff** for printing on a PostScript printer (type **postscript**), with requests for landscape orientation and the **gothic** character set, would enter the following command:

```
# lp -T troff -S gothic -y land -d any ↵
```

Then this filter would be invoked by the LP print service to convert the file as follows:

```
/usr/lib/lp/postscript/dpost -S gothic -o landscape
```

Example 3 —

You add the following option template to the previous example:

Options: MODES size=\([0-9]*\)x\([0-9]*\) = -h\1 -w\2

This template is used to convert a **MODES** option of the form

-y size=heightxwidth

into a pair of filter options,

-hheight -wwidth

So if a user gives the following command

```
# lp -y size=24x80 ↵
```

the **dpost** command would include the following options:

```
-h24 -w80
```

Command to enter

Once a filter definition is complete, enter one of the following commands to add the filter to the system.

```
# lpfilter -f filter-name -F file-name ↵  
# lpfilter -f filter-name - ↵
```

The first command gets the filter definition from a file, and the second command gets the filter definition from the standard input. A *filter-name* can be any string you choose, with a maximum of fourteen alphanumeric characters and underscores.

If you need to change a filter, just reenter one of the same commands. You need only provide information for those items that

must be changed; items for which you don't specify new information will stay the same.

Removing a filter

The LP print service imposes no fixed limit on the number of filters you can define. It is a good idea, however, to remove filters no longer applicable, to avoid extra processing by the LP print service which must examine all filters to find one that works in a given situation.

To remove a filter, enter the following command:

```
# lpfilter -f filter-name -x ↵
```

Examining a filter

Once you've added a filter definition to the LP print service, you can examine it by running the **lpfilter** command. The output of this command is the filter definition displayed in a format that makes it suitable as input. You may want to save this output in a file that you can use later to redefine the filter if you inadvertently remove the filter from the LP print service.

To examine a defined filter, enter one of the following commands:

```
# lpfilter -f filter-name -l ↵  
# lpfilter -f filter-name -l >file-name ↵
```

The first command presents the definition of the filter on your screen; the second command captures this definition in a file for future reference.

Restoring factory defaults

The software is shipped from the factory with default values set for all options. If, after changing these defaults, you want to reset them, execute the following command:

```
# lpfilter -f filter-name -i ↵
```

You can restore an individual filter by giving its name in place of *filter-name*, or you can restore everything by giving the name **all**.

A word of caution

Adding, changing, or deleting filters can cause print requests still queued to be canceled. This is because the LP print service

evaluates all print requests still queued, to see which are affected by the filter change. Requests that are no longer printable, because a filter has been removed or changed, are canceled (with notifications sent to the people who submitted them). There can also be delays in the responses to new or changed print requests when filters are changed, due to the many characteristics that must be evaluated for each print request still queued. These delays can become noticeable if there is a large number of requests that need to be filtered.

Because of this possible impact, you may want to make changes to filters during periods when the LP print service is not being used much.

Managing the printing load

Occasionally you may need to stop accepting print requests for a printer or move pending print requests from one printer to another. There are various reasons why you might want to do this, such as the following:

- the printer needs periodic maintenance
- the printer is broken
- the printer has been removed
- you've changed the configuration so that the printer is to be used differently
- too many large print requests are queued for one printer and should be spread around

If you are going to make a big change in the way a printer is to be used, such as stopping its ability to handle a certain form, changing the print wheels available for it, or disallowing some users from using it, print requests that are currently queued for printing on it will have to be moved or canceled. The LP print service will attempt to find alternate printers, but only if the user doesn't care which printer is to be used. Requests for a specific printer won't be automatically moved; if you don't move them first, the LP print service will cancel them.

If you decide to take a printer out of service, to change its configuration, or to lighten its load, you may want to move print requests off it and reject additional requests for it for awhile. To do so, use the **lpmove** and **reject** commands. If you do reject requests for a printer, you can accept requests for it later, by using the **accept** command.

Rejecting requests for a printer or class

To stop accepting any new requests for a printer or class of printers, enter the following command.

```
# reject -r "reason" printer-or-class-name ↵
```

You can reject requests for several printers or classes in one command by listing their names on the same line, separating the names with spaces. The "reason" will be displayed whenever anyone tries to print a file on the printer. You can omit it (and the **-r**) if you don't want to specify a reason.

Although the **reject** command stops any new print requests from being accepted, it will not move or cancel any requests currently queued for the printer. These will continue to be printed as long as the printer is enabled.

Accepting requests for a printer or class

After the condition that led to rejecting requests has been corrected or changed, enter one of the the following commands to start accepting new requests.

```
# accept printer-name ↵  
# accept class-name ↵
```

Again, you can accept requests for several printers or classes in one command by listing their names on the same line.

You will always have to use the **accept** command for a new printer or class after you have added it, because the LP print service does not initially accept requests for new printers or classes.

Moving requests to another printer

If you specify **-d any** when you run the **lp** command to queue a job, the print service schedules the job for a particular printer. If another becomes available first, the job is sent to the latter printer. If a job is scheduled for a given printer and you run **lpmove** to get jobs off that printer, that job will be moved off and the destination will change from **any** to the printer you've specified on the **lpmove** command line. Users may not have intended this side effect. If not, run the following command:

```
# lp -i request-ID -d any ↵
```

This command will change the destination for the requested job to the original destination: **any** (that is, any available printer).

If you have to move requests from one printer or class to another, enter one of the following commands

```
# lpmove request-id printer-name ↵  
# lpmove printer-name1 printer-name2 ↵
```

You can give more than one request ID before the printer name in the first command.

The first command above moves the listed requests to the printer *printer-name*. The second command tries to move *all* requests currently queued for *printer-name1* to *printer-name2*. If some requests cannot be printed on the new printer, they will be left in the queue for the original printer. When the second command is used, the LP print service also stops accepting requests for *printer-name1* (the same result you would obtain by running the command **reject** *printer-name2*).

Examples

Here are some examples of how you might use these three commands.

Example 1

You've decided it is time to change the ribbon and perform some preventive maintenance on printer **lp1**. First, to prevent the loss of print requests already queued for **lp1**, you move all requests from printer **lp1** to printer **lp2**.

```
# lpmove lp1 lp2 ↵
```

After the requests are moved, make sure the LP print service does not print any more requests on **lp1** by disabling it.

```
# disable lp1 ↵
```

Now you may physically disable the printer and start working on it.

Example 2

You've finished changing the ribbon and doing the other work on **lp1**; now it's time to bring it back into service. Execute the following commands in any order:

```
# accept lp1 ↵
```

```
# enable lp1 ↵
```

See Chapter 8 for details on enabling printers.

Example 3

You notice that someone has queued several large files for printing on the printer **laser1**. Meanwhile **laser2** is idle because no one has

queued requests for it. Move the two biggest requests (**laser1-23** and **laser1-46**) to **laser2**, and reject any new requests for **laser1** for the time being.

```
# lpmove laser1-23 laser1-46 laser2 ↵
# reject -r"too busy—will reopen later" laser1 ↵
```

Managing queue priorities

The LP print service provides a simple priority mechanism that users can use to adjust the position of a print request in the queue. Each print request can be given a priority level by the user who submits it; this is a number from 0 to 39, with *smaller* numbers indicating *higher* levels of priority. Requests with higher priority (smaller numbers) are placed ahead of requests with lower priority (larger numbers).

Thus, for example, a user who decides that her print request is of low priority can assign it a larger value when she submits the file for printing. Another user who decides that his print request is of high priority can assign it a smaller value when he submits the file for printing.

A priority scheme this simple would not work if there were no controls on how high one can set the priority. You can define the following characteristics in this scheme:

- Each user can be assigned a priority limit. One cannot submit a print request with a priority higher than his or her limit, although one can submit a request with a lower priority.
- A default priority limit can be assigned for the balance of users not assigned a personal limit.
- A default priority can be set. This is the priority given print requests to which the user does not assign a priority.

By setting the characteristics according to your needs, you can prevent lower priority printing tasks (such as regular printing by most staff members) from interfering with higher priority printing tasks (such as payroll check printing by the accounting staff).

You may find that you want a critical print request to print ahead of any others, perhaps even if it has to preempt the currently printing request. You can have the LP print service give immediate handling to a print request, and you can have it hold another print request. This will allow the first request to be printed and will delay the second print request until you allow it to be resumed.

The **lpusers** command lets you assign both priority limits for users and priority defaults. In addition, you can use the **lp -irequest-id**

-Hhold and **lp -irequest-id -Himmediate** commands to put a request on hold or to move it up for immediate printing, respectively. These commands are discussed in detail below.

Setting priority limits

To set a user's priority limit, enter the following command.

```
# lpusers -q priority-level -u username ↵
```

You can set the limit for a group of users by listing their names after the **-u** option. Separate multiple names with a comma or space (enclose the list in quotes if you use a space, though). The argument *priority-level* is a number from 0 to 39. As mentioned before, the lower the number the higher the priority, or, in this case, the priority limit.

If you want to set a priority limit for the remaining users, enter the following command:

```
# lpusers -q priority-level ↵
```

This sets the default limit; the default applies to those users for whom you have not set a personal limit, using the first **lpusers** command.

If you later decide that someone should have a different priority limit, just reenter the first command above with a new limit. Or, if you decide that the default limit is more appropriate for someone who already has a personal limit, enter the following command:

```
# lpusers -u username ↵
```

Again, you can do this for more than one user at a time by including a list of names. Using the **lpusers** command with just the **-u** option removes users' personal priority limits and puts the default limit into effect for those users.

Setting a default priority

To set the default priority (the priority level assigned to print requests submitted without a priority), use the following command:

```
# lpusers -d priority-level ↵
```

Don't confuse this default with the default limit. This default is applied when a user doesn't specify a priority level; the default limit

is applied if you haven't assigned a limit for a user—it is used to limit the user from requesting too high a priority. If the default priority is greater than the limit for a user, the limit is used instead.

If you do not set a default priority, the LP print service will use a default of 20.

Examining the priority limits and defaults

You can examine all the settings you have assigned for priority limits and defaults by entering the following command.

```
# lpusers -l ↵
```

Moving a request around in the queue

Once a user has submitted a print request, you can move it around in the queue to some degree:

- you can adjust the priority to any level, regardless of the limit for the user (who may adjust it only up to his or her limit)
- both you and the user can put it on hold and allow other requests to be printed ahead of it
- you can put it at the head of the queue for immediate printing

Use the `lp(1)` command to do any of these tasks.

Changing the priority for a request

If you want to change the priority of a particular request that is still waiting to be printed, you can assign a new priority level to it. By doing so, you can move it in the queue so that it is ahead of lower priority requests, and behind requests at the same level or of higher priority. The priority limit assigned to the user (or the default priority limit) has no effect because, as the administrator, you can override this limit.

Enter the following command to change the priority of a request.

```
# lp -i request-ID -q new-priority-level ↵
```

You can change only one request at a time with this command.

Putting a request on hold

Any request that has not finished printing can be put on hold. This will stop its printing, if it is currently printing, and keep it from

printing until you resume it. A user may also put his or her own request on hold and then resume it, but may not resume a print request that has been put on hold by the administrator.

To place a request on hold, enter the following command:

```
# lp -i request-ID -H hold ↵
```

Enter the following command to resume the request:

```
# lp -i request-ID -H resume ↵
```

Once resumed a request will continue to move up the queue and will eventually be printed. If printing had already begun when you put it on hold, it will be the next request printed.

Moving a request to the head of the queue

You can move a print request to the head of the queue where it will be the next one eligible for printing. If you want it to start printing immediately but another request is currently being printed, you may interrupt the first request by putting it on hold, as described above.

Enter the following command to move a print request to the head of the queue:

```
# lp -i request-ID -H immediate ↵
```

Only you, as the administrator, can move a request in this way; regular users cannot use the **-H immediate** option. If you set more than one request for immediate printing, the requests will be printed in the reverse order set; that is, the request moved to the head of the queue most recently will be printed first.

Starting and stopping the LP print service

Under normal operation, you should never have to start or stop the LP print service manually. The system starts it each time the system boots, and stops it each time the system stops. If, however, you need to stop the LP print service without stopping the entire system, do so by following the procedure described below.

Stopping the LP print service will cause all printing to cease within seconds. Any print requests that have not finished printing will be printed in their entirety after the LP print service is restarted. The

printer configurations, forms, and filters in effect when the LP print service is stopped will be restored after it is restarted. To start and stop the LP print service manually, you must be logged in as either the user **lp** or the super-user (**root** or **sysadm**).

Manually stopping the print service

To stop the LP print service manually, enter the following command:

```
# lpshut ↵
```

The message

```
Print services stopped.
```

appears, and all printing will cease within a few seconds. If you try to stop the LP print service when it is not running, you will see the message

```
Print services already stopped.
```

For additional cleanup and error recovery, you should use the following command to stop the print service:

```
# sysadm stoplp ↵
```

This executes **admlp** scripts which provide additional cleanup features.

Manually starting the print service

To restart the LP print service manually, enter the following command:

```
# lpsched ↵
```

The message:

```
Print services started.
```

will appear. It may take a minute or two for the printer configurations, forms, and filters to be reestablished, before any saved print requests start printing. If you try to restart the LP print service when it is already running, you will see the message

```
Print services already active.
```

For additional cleanup and error recovery, you should use the following command to start the print service:

```
# sysadm startlp ↵
```

This executes **admlp** scripts which provide additional cleanup features.

The LP print service does not have to be stopped to change printer configurations or to add forms or filters.

Managing the LP print service logs

The LP print service has several logs that accumulate information on various LP services. By default, the LP print service does not remove or truncate these logs; therefore, you need to make sure they do not consume too much disk space. The easiest way to manage these logs is by submitting the **cron** jobs provided for the purpose in the LP print service prototype **crontab** file, **/admin/crontabs/lp.proto**. For more information on submitting **cron** jobs, see the manual pages for **cron(1M)** and **crontab(1)**.

Located in **/var/spool/lp/logs**, the LP print service logs are **lpNet**, **lpsched**, and **requests**. The following sections elaborate.

The lpNet log

The LP system runs a process called **lpNet**, which handles communication with LP systems running on other systems in your network. The **lpNet** process logs its activities to the file **/var/lp/logs/lpNet**.

As the **lpNet** file grows, you should clean it out occasionally. The easiest way to do this is with the appropriate **cron** job from **/admin/crontabs/lp.proto**. The job for this purpose appears below.

```
17 3 * * 0 /usr/sbin/agefile -c4 /var/lp/logs/lpNet
```

This **cron** job ages the **lpNet** log so that every week starts a new **lpNet** log, and no log remains on the system for more than five weeks. This job runs every Sunday at 3:17 am.

The lpsched log

The LP print service keeps a log for scheduler-related events. This log is **/var/spool/lp/logs/lpsched**.

As with the **lpNet** log, the best way to control the size of the **lpsched** log is by submitting the appropriate **cron** job from **/admin/crontabs/lp.proto**. The job for this purpose appears below.

```
15 3 * * 0 /usr/sbin/agefile -c4 /var/lp/logs/lpsched
```

This **cron** job ages the **lpsched** log so that every week starts a new **lpsched** log, and no log remains on the system for more than five weeks. This job runs every Sunday at 3:15 am.

The requests log

The **/var/spool/lp/logs/requests** file accumulates information about each completed request. Initially, the directories **/var/spool/lp/tmp/system** and **/var/spool/lp/requests/system** contain files that describe each request that has been submitted to the LP print service. Each request has two files (one in each directory) that contain information about the request. The information is split to put more sensitive information in the **/var/spool/lp/requests/system** directory where it can be kept secure: the request file in the **/var/spool/lp/tmp/system** is safe from all except the user who submitted the request, while the file in **/var/spool/lp/requests/system** is safe from all users, including the submitting user.

These files remain in their directories only as long as the request is in the queue. Once the request is finished, the information in the files is combined and appended to the file **/var/lp/logs/requests**.

Log structure —

The **requests** log has a simple structure that makes it easy to extract data from it using common shell commands. Requests are listed in the order they are printed, and are separated by lines showing their request IDs. Each line below the separator line is marked with a single letter that identifies the kind of information contained in that line. Each letter is separated from the data by a single space. See the following list for details.

- =** This is the separator line. It contains the request ID, the user and group IDs of the user, the total number of bytes in the original (unfiltered) files, and the time when the request was queued. These items are separated by commas and are in the order just named. The user ID, group ID, and sizes are preceded by the words **uid**, **gid**, and **size**, respectively.
- C** The number of copies printed.

Continued

- D** The printer or class destination or the word **any**.
- F** The name of the file printed. This line is repeated for each file printed; files were printed in the order given.
- f** The name of the form used.
- H** One of three types of special handling: **resume**, **hold**, and **immediate**. The only useful value found in this line will be **immediate**.
- N** The type of alert used when the print request was successfully completed. The type is the letter **M** if the user was notified by mail, or **W** if the user was notified by a message to his or her terminal.
- O** The **-o** options.
- P** The priority of the print request.
- p** The list of pages printed.
- r** This single letter line is included if the user asked for *raw* processing of the files (the **-r** option of the **lp** command).
- S** The character set or print wheel used.
- s** The outcome of the request, shown as a combination of individual bits expressed in hexadecimal form. While several bits are used internally by the print service, the most important bits are listed below:
 - 0x0004 Slow filtering finished successfully.
 - 0x0010 Printing finished successfully.
 - 0x0040 The request was canceled.
 - 0x0100 The request failed filtering or printing.
- T** The title placed on the banner page.
- t** The type of content found in the files.
- U** The name of the user who submitted the print request.
- x** The slow filter used for the request.
- Y** The list of special modes to give to the filters used to print the request.
- z** The printer used for the request. This will differ from the destination (the **D** line) if the request was queued for any printer or a class of printers, or if the request was moved to another destination by the LP print service administrator.

Cleaning out the requests log —

The LP print service does not remove the **requests** file; therefore, the file will grow indefinitely if you do not remove it or shorten it

occasionally. The best way to do this is with the **cron** job provided in the LP print service's prototype **crontab** file, **/admin/crontabs/lp.proto**. The prototype file contains three **cron** jobs. The one for managing the **requests** file appears below.

```
13 3 * * * cd /var/lp/logs; if [ -f requests ]; then \  
  /bin/mv requests xyzzy; /bin/cp xyzzy requests; >xyzzy; \  
  /usr/lbin/agefile -c2 requests; /bin/mv xyzzy requests; fi
```

This jobs appears as one line in the **crontab** file, but it is split into several lines here for readability.

What this entry does, briefly, is *age* the file, changing the name to **requests-1**, and moving the previous day's copy to **requests-2**. The number **2** in the **-c** option to the **agefile** program keeps the log files from the previous two days, discarding older log files. By changing this number you can change the amount of information saved. On the other hand, if you want the information to be saved more often, or if you want the file to be cleaned out more often than once a day, you can change the time when the **crontab** entry is run by changing the first two numbers. The current values, **13** and **3**, cause cleaning up to be done at 3:13 am each day.

The default **crontab** entry supplied is sufficient to keep the old print request records from accumulating in the spooling file system. You may want to condense information in the request log to produce a report on the use of the LP print service, or to aid in generating accounting information. You can produce a different script that examines the file and extracts information just before the clean up procedure.

Managing PostScript printers

PostScript is a general purpose programming language, like C or Pascal. In addition to providing the usual features of a language, however, PostScript allows a programmer to specify the appearance of both text and graphics on a page.

A PostScript printer is a printer equipped with a computer that runs an interpreter for processing PostScript language files. When a PostScript printer receives a file, it runs that file through the interpreter and then prints it. Unless special provisions have been made by the manufacturer, files submitted to a PostScript printer must be written in the PostScript language.

Why would you want to use a PostScript printer? PostScript provides excellent facilities for managing text and graphics and combining them. Graphics operators facilitate the construction of geometric figures which can then be positioned and scaled with any

orientation. The text capabilities allow the user to specify a number of different fonts that can be placed on a page in any position, size, or orientation. Because text is treated as graphics, text and graphics are readily combined. Moreover, the language is resolution and device independent, so that draft copies can be proofed on a low-resolution device and the final version printed in higher resolution on a different device.

Applications that support PostScript, including word-processing and publishing software, will create documents in the PostScript language without intervention by the user. Thus, it is not necessary to know the details of the language to take advantage of its features. However, standard files that many applications produce cannot be printed on a PostScript printer because they are not described in the language. The LP print service provides optional filters to convert many of these files to PostScript so that users may take advantage of PostScript and continue to use their standard applications such as **troff**.

How to use a PostScript printer

When the PostScript printers and filters have been installed, the LP subsystem manages PostScript files like any others. If **psfile** is a file containing a PostScript document and **psprinter** has been defined as a PostScript printer, the command

```
# lp -d psprinter -Tpostscript psfile ↵
```

will schedule the print request and manage the transmission of the request to the PostScript printer.

Support of non-PostScript print requests

A PostScript printer may not be able to interpret every kind of file that an application sends to it. The following list names a few of the formats that your printer may not be able to accommodate.

troff	Print a troff text file.
simple	Print an ASCII text file.
dmd	Print the contents of a bit-mapped display (for example, an AT&T 630).
tek4014	Print files formatted for a Tektronix 4014 device.
daisy	Print files intended for a daisy-wheel printer (for example, a Diablo 630).

plot Print plot-formatted files

Optional filters are provided with the LP print service to translate print requests with these formats to the PostScript language. For example, to convert a file containing ASCII or **troff** code to PostScript code, the filter takes that text and writes a program around it, specifying printing parameters such as fonts and the layout of the text on a page.

Once the PostScript filters are installed, they will be invoked automatically by the LP print service when a user specifies a content-type for a print request with the **-T** option. For example,

```
# lp -d psprinter -T simple report2 ↵
```

will automatically convert the ASCII file **report2** (a file with an ASCII or **simple** format) to PostScript when the destination printer *psprinter* has been defined to the system as a PostScript printer. Note that **-T simple** is the default content type and is used here for example purposes only.

Additional PostScript capabilities provided by filters

The filters previously described also take advantage of PostScript capabilities to provide additional printing flexibility. Most of these features may be accessed through the mode option (invoked by the **-y** option) to the **lp** command. These filters allow you to use several unusual options for your print jobs. The following list describes these options and shows the option you should include on the **lp** command line for each one. Most of these options are provided by the **postprint** filter. See the **postprint(1)** manual page for more information.

- y reverse** Reverse the order in which pages are printed.
- y landscape** Change the orientation of a physical page from portrait to landscape.
- y x=xnumber,y=ynumber**
Change the default position of a logical page on a physical page by moving the origin.
- y group=number**
Group multiple logical pages on a single physical page.
- P number** Select, by page numbers, a subset of a document to be printed

If these filters are to be used with an application that creates PostScript output, make sure that the application conforms to the

PostScript file structuring comments. In particular, the beginning of each PostScript page must be marked by the comment

```
%%Page:label ordinal
```

where *ordinal* is a positive integer that specifies the position of the page in the sequence of pages in the document.

For example, say you have a file called **report2** that has a content type **simple** (meaning that the content of this file is in ASCII format). You want to print six pages of this file (pages 4-9) in landscape mode (that is, sideways on the page) with two logical pages on each physical page. Because one of the printers on your system (**psprinter**) is a PostScript printer, you can do this, by entering the following command:

```
# lp -d psprinter -T simple -P 4-9 -y landscape,group=2 report2 ␣
```

In addition, the LP print service offers a special filter that can print a gray-scale representation of a matrix. (A gray-scale matrix is a matrix in which each cell is colored one of seven shades of gray to indicate the value of the cell. Darker shades correspond to larger values.) To print a gray-scale representation, specify **matrix** as the content type of your source file by giving the **-T matrix** option.

The administrator's duties

Support of PostScript printers is similar to support of other printers, in that the printers must be defined to the system with the **lpadm** command and the appropriate software must be installed to manage them. PostScript printers may require some additional effort in supporting fonts.

IMPORTANT: A print job submitted to a serial PostScript printer on an AViiON 500 series system may be truncated. This problem may occur with Data General model 6646T, 6779T, 6772, and 6773 printers that are connected by cable 15340E, through a VDA cluster box (VDC/8p/16) or a VAC 16. This will be fixed in a future release. Meanwhile, to work around the problem, add **clocal** to the `Stty` options: query with the **sysadm** Device -> Printer -> Device -> Modify operation.

Installing and maintaining PostScript printers

PostScript printers, like other printers, are installed with the **lpadm** command. The content type of a PostScript printer must be consistent with the content type used in PostScript filters. Consequently, you should install serial PostScript printers with a

content type of **PS**, causing LP to use the **postio** filter to communicate with the printer. When installing a parallel printer, however, specify a content type of **postscript** instead, causing LP not to use the **postio** filter. The **postio** filter requires a bidirectional communication connection, which can only be supplied by serial lines.

The content type accepted by the printer is specified with the **-I** option in the **lpadm** command. In addition, you must tell the LP print service which fonts are available on the printer. To do so, enter this information in the font list for each printer.

Maintaining PostScript filters

PostScript filters provided with this release are pre-installed. This section contains specific information about the location and function of these filters. In certain circumstances, you may find it helpful to change existing filter descriptions.

PostScript filters are contained in the directory **/usr/lib/lp/filter/postscript**.

There are two types of filters: fast filters and slow filters. For a definition of these types, see the **lpfilter(1M)** manual page and “Defining a Filter” earlier in this chapter.

A prerequisite of communication between any system and a serial PostScript printer is the presence of the **postio** filter on the system. This program is the only mandatory PostScript filter for serial PostScript printers. Table 9–5 lists the filters that allow other types of documents to be translated to PostScript and to be printed on a PostScript printer.

Table 9–5 PostScript filters

File content type	Filter
simple	postprint
troff	dpost
daisy (as for Diablo 630)	postdaisy
dmd (as for AT&T 630)	postdmd
tek4014 (Tektronix)	posttek
plot	postplot

Table 9–6 lists the filters that perform special functions:

Table 9–6 Special function filters

Function	Filter
Communicate with printer	postio
Download fonts	download
Reverse or select pages	postreverse
Matrix gray scales	postmd

Installing and maintaining PostScript fonts

One of the advantages of PostScript is its ability to manage fonts. Fonts are stored in outline form, either on the printer or on a host that communicates with a printer. When a document is printed, the PostScript interpreter generates each character as needed (in the appropriate size) from the outline description of it. If a font required for a document is not stored on the printer being used, it must be transmitted to that printer before the document can be printed. This transmission process is called “downloading fonts.”

Fonts are stored and accessed in several ways.

- Fonts may be stored permanently on a printer. These “printer resident” fonts may be installed in ROM on the printer by the manufacturer. If the printer has a disk, fonts may be installed on that disk by you (that is, by the print service administrator). Most PostScript printers are shipped with thirty-five standard fonts.
- A font may be “permanently-downloaded” by being transmitted to a printer with a PostScript “exitserver” program. A font downloaded in this way will remain in the printer’s memory until the printer is turned off. Memory allocated to this font will reduce the memory available for PostScript print requests. Use of exitserver programs requires the printer system password and may be reserved for the printer administrator. This method is useful when there is continual use of a font by the majority of print requests serviced by that printer.
- Fonts may be prefixed to a user’s print request by the user, and be transmitted as part of the user’s print request. When the user’s document has been printed, the space allocated to the font is freed for other print requests. The font is stored in the user’s directory. This method is preferable for fonts with more limited usage.
- Fonts may be stored on a system shared by many users. These fonts may be described as “host-resident.” This system may be a server for the printer or may be a system connected to the printer by a network. Each user may request fonts in the document to be

printed. This method is useful when there are a large number of available fonts or there is not continual use of these fonts by all print requests. If the fonts will be used only on printers attached to a server, they should be stored on the server. If the fonts are to be used by users on one system, who may send jobs to multiple printers on a network, they may be stored on the users' system.

The LP print service allows you to manage fonts in any of three ways. It provides a special download filter to manage fonts using the last method described above.

Managing printer-resident fonts

Most PostScript printers come equipped with fonts resident in the printer ROM. Some printers have a disk on which additional fonts are stored. When a printer is installed, the list of printer-resident fonts should be added to the font-list for that printer. These lists are kept in the printer administration directories. For a particular printer, this list is contained in the file

/etc/lp/printers/printer-name/residentfonts

where *printer-name* is the name of the printer.

When fonts are permanently downloaded to the printer, the font names should be added to this file. Host-resident fonts must reside on the host to which the printer is attached. Using System V Release 4 **lpNet** printing, fonts are not transferred from a local printer to the remote host. This list should include fonts which reside on that system and are available for downloading to the printer. These files must be edited manually; that is, with the help of a text editor such as **vi(1)**.

For example, the **residentfonts** file for the DG Model 6771 PostScript printer with the 35 font option would include:

AvantGarde-Book
AvantGarde-BookOblique
AvantGarde-Demi
AvantGarde-DemiOblique
Bookman-Demi
Bookman-DemiItalic
Bookman-Light
Bookman-LightItalic
Courier
Courier-Bold
Courier-BoldOblique
Courier-Oblique
Helvetica
Helvetica-Bold
Helvetica-BoldOblique
Helvetica-Narrow
Helvetica-Narrow-Bold
Helvetica-Narrow-Bold
Helvetica-Narrow-BoldOblique
Helvetica-Narrow-Oblique
Helvetica-Oblique
NewCenturySchlbk-Bold
NewCenturySchlbk-BoldItalic
NewCenturySchlbk-Italic
NewCenturySchlbk-Roman
Palatino-Bold
Palatino-BoldItalic
Palatino-Italic
Palatino-Roman
Symbol
Times-Bold
Times-BoldItalic
Times-Italic
Times-Roman
ZapfChancery-MediumItalic
ZapfDingbats

Installing and maintaining host-resident fonts

Some fonts will be resident on the host and transmitted to the printer as needed for particular print requests. As the administrator, it's your job to make PostScript fonts available to all the users on a system. To do so, you must know how and where to install these fonts, using the guidelines described previously. Because fonts are requested by name and stored in files, LP keeps a map file with the correspondence between font names and the names of the files that contain the fonts. Both of these must be updated when fonts are installed on the host.

Install host-resident PostScript fonts by doing the following:

- Copy the font file to the appropriate directory.
- Add to the map table the name of the font and the name of the file in which it resides.

Where are fonts stored? —

The fonts available for use with PostScript printers reside in directories called `/usr/share/lib/hostfontdir/typeface/font`. Replace *typeface* with a typeface name such as `palatino` or `helvetica`. Replace *font* with `bold`, `italic`, and so forth.

Adding an entry to the map table —

Also within the `hostfontdir` directory, you must create and maintain a map table file, called `map`, which shows the correspondence between the name assigned to each font by the foundry (the company that created the font) and the name of the file in which that font resides. For example, to map the font called “XYZ-SemiBold,” add the following line to the map table file:

```
XYZ-SemiBold /usr/share/lib/hostfontdir/xyz/XYZ-SemiBold
```

The font file, `XYZ-SemiBold`, must be stored in the above directory. The font vendor will provide the suitable PostScript program for storing the font into a font dictionary. An example entry would be:

```

%!
serverdict begin 0 exitserver
14 dict begin
/FontInfo 10 dict dup begin
/Notice (Copyright XYZ Corporation 1992) \
All rights reserved. This product is licensed, not sold, \
and may not be reproduced with out the consent of \
XYZ Corporation.) readonly def
  /FullName (Xyz Semi Bold)def
  /FamilyName(Xyz)def
  /Weight(Semi Bold (Demi))def
  /version(1.1)def
  /ItalicAngle 0 def
  /isFixedPitch false def
  /UnderlinePostion -175 def
  /UnderlineThickness 41 def
end readonly def
/FontName /XYZ-SemiBold def
/Encoding StandardEncoding def
/PaintType 0 def
/FontType 1 def
/FontMatrix [0.001 0 0 0.001 0 0] readonly def
/FontBox {-155 -214 1006 865} readonly def
/Unique ID nnnnnnnnn def
currentdict end
currentfile eexec
  font vector description
cleartomark

```

Once this entry exists in the map table file on your system, your users will be able to have an XYZ-SemiBold font used in their print jobs. When they submit a file containing a request for this font, the LP print service prepends a copy of the following file to the user's file before sending it to the printer.

```
/usr/share/lib/hostfontdir/xyz/XYZ-SemiBold
```

Downloading host-resident fonts

The creators of the PostScript language anticipated that users would want to download fonts to printers. For this purpose, they defined a standard set of *structuring conventions* for PostScript programs. The download filter relies on these structuring conventions to determine which fonts must be downloaded. See your PostScript documentation for more information.

When the LP print service receives a request for a job that requires fonts not loaded on the printer, it forwards that request to a filter that downloads fonts.

The download filter does the following:

- Searches the PostScript document to determine which fonts have been requested. These requests are documented with the following PostScript structuring comments in the header comments:

```
%%DocumentFonts: font1 font2 ...
```

- Searches the list of fonts resident on that printer to see if the requested font must be downloaded.
- If the font is not resident on the printer, searches the host-resident font directory (by getting the appropriate file name from the map table) to see if the requested font is available.
- If the font is available, prepends the file for that font to the file to be printed.
- Sends the font definition file and the source file (the file to be printed) to the PostScript printer.

For example, a document may have the following font request:

```
%%Documentfonts: Helvetica XYZ-SemiBold
```

In our host-resident list above, the font, XYZ-SemiBold, is not listed. However being listed in the map table, XYZ-SemiBold is available for downloading.

Permanently downloaded fonts

If you frequently request a common set of fonts, you might wish to permanently download them to the printer. Note that these fonts will decrease the amount of memory available on the printer and must be downloaded each time the power is cycled on the printer.

For example, follow these steps to permanently download the font, ZZZ-Bold:

1. Make sure an entry in the resident font file, **/etc/lp/printers/printer-name/residentfonts** exists for ZZZ-Bold. This entry prevents the font being downloaded for each print request.
2. Initialize the printer by issuing the command

```
lp /usr/share/lib/hostfontdir/zzz/ZZZ-Bold ↵
```

to initially download the font. The font file should contain PostScript commands similar to the following. This header should be supplied by the font vendor.

```
%!  
serverdict begin 0 exitserver  
14 dict begin  
/FontInfo 10 dict dup begin  
    /Notice (Copyright ZZZ Corporation 1992) \  
All rights reserved. This product is licensed, not  
sold, \  
and may not be reproduced with out the consent of \  
ZZZ Corporation.) readonly dev  
    /FullName (Zzz Semi Bold)def  
    /FamilyName(Zzz)def  
    /Weight(Semi Bold (Demi))def  
    /version(1.1)def  
    /ItalicAngle 0 def  
    /isFixedPitch false def  
    /UnderlinePostion -175 def  
    /UnderlineThickness 41 def  
end readonly def  
/FontName /ZZZ-SemiBold def  
/Encoding StandardEncoding def  
/PaintType 0 def  
/FontType 1 def  
/FontMatrix [0.001 0 0 0.001 0 0] readonly def  
/FontBox {-155 -214 1006 865} readonly def  
/Unique ID nnnnnnnnnn def  
currentdict end  
currentfile eexec  
    font vector description  
cleartomark
```

Customizing the print service

Although the LP print service has been designed to be flexible enough to handle most printers and printing needs, it doesn't handle every possible situation. You may buy a printer that doesn't quite fit into the way the LP print service handles printers, or you may have a printing need that the standard features of the LP print service don't accommodate.

You can customize the LP print service in a few ways. This section tells you how you can do the following:

- Adjust the printer port characteristics
- Adjust the **terminfo** database
- Write an interface program
- Write a filter

- Do character set translations

The diagram in Figure 9–1 gives an overview of the processing of a print request.

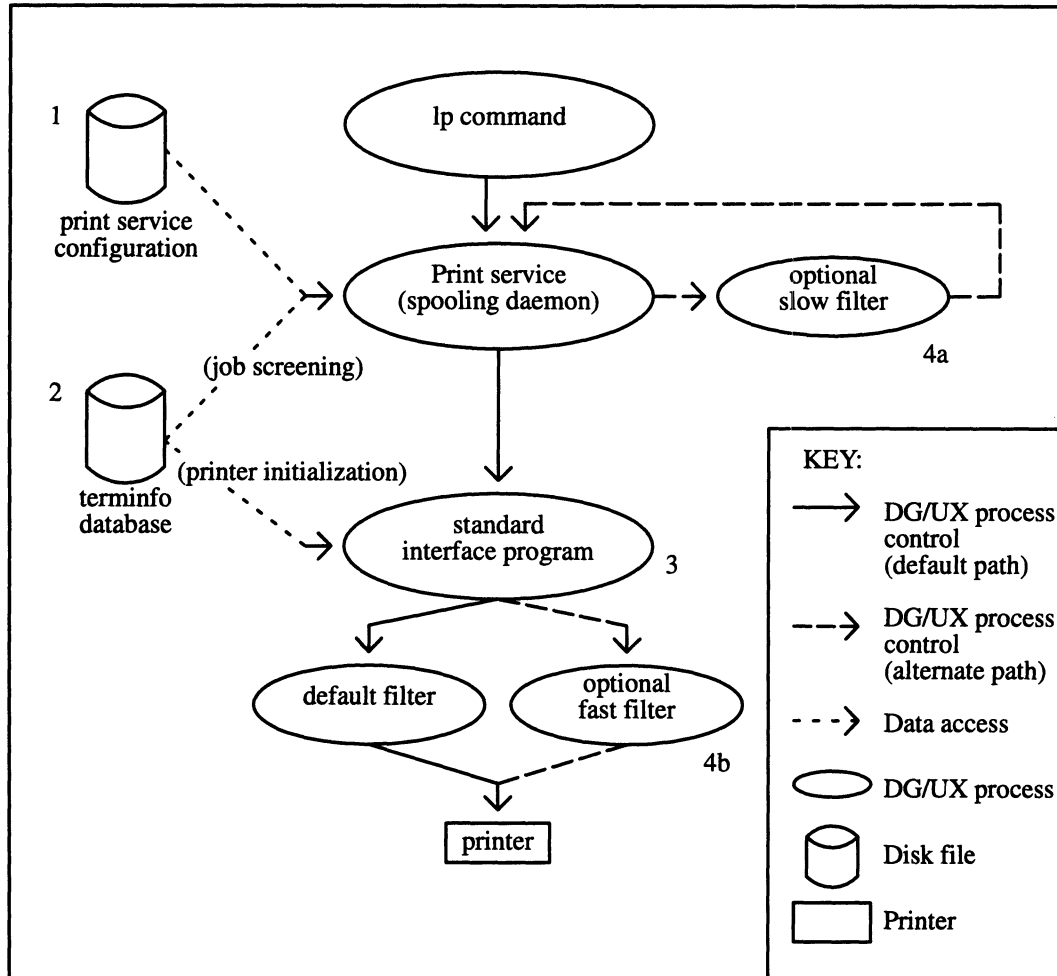


Figure 9–1 How LP processes print a request

Each print request is sent to a spooling daemon that keeps track of all requests. The daemon, which is created when you start the LP print service, is also responsible for keeping track of the status of printers and slow filters; when a printer finishes printing a user's file, the daemon starts it printing another request (if there is one queued).

To customize the print service, adjust or replace some of the pieces shown in Figure 9–1. (The numbers are keyed to the diagram.)

1. For most printers, you need only change the printer configuration stored on disk. The earlier sections of this chapter explain how to do this. Configuration data that are relatively dependent on the printer include the printer port characteristics: baud rate, parity, and so on.

2. For a printer that is not represented in the **terminfo** database, you can add a new entry that describes its capabilities. The **terminfo** database is used in two parallel capacities: screening print requests to ensure that those accepted can be handled by the desired printer, and setting the printer in a state where it is ready to print a request.

For instance, if the **terminfo** database does not contain an entry for a printer capable of setting a page length requested by a user, the spooling daemon will reject the request. On the other hand, if it does contain an entry for such a printer, then the same information will be used by the interface program to initialize the printer.

3. For particularly difficult printers, or if you want to add features not provided by the delivered LP print service, you can change the standard interface program. This program is responsible for managing the printer: it prints the banner page, initializes the printer, and invokes a filter to send copies of a user's files to the printer.
- 4a. and 4b. To provide a link between the applications used on your system and the printers, you can add slow and fast filters. Each type of filter can convert a file into another form, mapping one set of escape sequences into another, for instance, and can provide special setup by interpreting print modes requested by a user. Slow filters are run separately by the daemon, to avoid tying up a printer. Fast filters are run so their output goes directly to the printer; thus they can exert control over the printer.

Adjusting the printer port characteristics

You should make sure that the printer port characteristics set by the LP print service match the printer communication settings. The standard printer port settings have been designed to work with typical files and many printers, but they won't work with all files and printers. This isn't really a customizing step, because a standard feature of the LP print service is to allow you to specify the port settings for each printer. However, it's an important step in getting your printer to work with the LP print service, so it's described in more detail here.

When you add a new printer, read the documentation that comes with it so that you understand what it expects from the host (the LP print service). Then read the manual page for **stty(1)**. It summarizes the various characteristics that can be set on a terminal or printer port.

Only some of the characteristics listed in the **stty(1)** manual page are important for printers. The ones likely to be of interest to you appear in Table 9-7 (but you should still consult the **stty(1)** manual page for others).

Table 9–7 stty options related to printers

stty Option	Description
evenp	Send even parity in the 8th bit
oddp	Send odd parity in the 8th bit
–parity	Do not generate parity; send all 8 bits unchanged
110 – 38400	Set the communications speed to this baud rate
ixon	Enable XON/XOFF (also known as START/STOP or DC1DC3) flow control
–ixon	Turn off XON/XOFF flow control
–opost	Do not do any “output–processing” according to the settings listed below
opost	Do “output post–processing” according to the settings listed below
onlcr	Send a carriage return before every linefeed
–onlcr	Do not send a carriage return before every linefeed
–tabs	Change tabs into an equivalent number of spaces
tabs	Do not change tabs into spaces

When you have a set of printer port characteristics you think should apply, adjust the printer configuration as described in Chapter 8. You may find that the default settings are sufficient for your printer.

Adjusting the terminfo database

The LP print service relies on a standard interface and the **terminfo** database to initialize each printer and establish a selected page size, character pitch, line pitch, and character set. Thus, it is usually sufficient to have the correct entry in the **terminfo** database to add a new printer to the LP print service. Several entries for Data General printers and other popular printers are delivered in the standard **terminfo** database.

Each printer is identified in the **terminfo** database with a short name; this kind of name is identical to the kind of name used to set the **TERM** shell variable. The complete list of supported terminal and printer names are in the **terminfo** database. To view the entire list, use this command line:

```
# ls -CR /usr/share/lib/terminfo/* | more ↵
```

If you cannot find a **terminfo** entry for your printer, you should add one. If you do not, you may still be able to use the printer with the LP print service, but you will not have the option of automatic selection of page size, pitch, and character sets, and you may have trouble keeping the printer set in the correct modes for each print request. Another option to follow, instead of updating the **terminfo** entry, is to customize the interface program used with the printer. (See the next section for details on how to do this.)

There are hundreds of items that can be defined for each terminal or printer in the **terminfo** database. However, the LP print service uses fewer than 50 of these. Table 9–8 lists some of the items that need to be defined (as appropriate for the printer) to add a new printer to the LP print service.

Table 9–8 terminfo items relevant to printers

terminfo item	Description
Booleans	
cpix	Changing character pitch changes resolution
daisy	Printer needs operator to change character set
lpix	Changing line pitch changes resolution
Numbers	
bufsz	Number of bytes buffered before printing
cols	Number of columns in a line
cps	Average print rate in characters per second
it	Tabs initially ever # spaces
lines	Number of lines on a page
orc	Horizontal resolution in units per character
orhi	Horizontal resolution in units per inch
orl	Vertical resolution in units per inch
orvi	Vertical resolution in units per inch
Strings	
chr	Change horizontal resolution
cpj	Change number of characters per inch
cr	Carriage return
csnm	List of character set names
cud1	Down one line
cud	Move carriage down # columns

Continued

Table 9–8 terminfo items relevant to printers

terminfo item	Description
Strings	
cuf	Move carriage right # columns
cuf1	Carriage right
cvr	Change vertical resolution
ff	Page eject
hpa	Horizontal position absolute
ht	Tab to next 8–space tab stop
if	Name of initializing file
ipro	Pathname of initialization program
is1	Printer initialization string
is2	Printer initialization string
is3	Printer initialization string
lpi	Change number of lines per inch
mgc	Clear all margins (top, bottom, and sides)
rep	Repeat a character # times
rwidm	Disable double wide printing
scs	Select character set
scsd	Start definition of a character set
slines	Set page length to # lines
smgb	Set bottom margin at current line
smgbp	Set bottom margin
smgl	Set left margin at current column
smglp	Set left margin at current column
smgt	Set top margin at current line
smgtp	Set top margin
swidm	Enable double wide printing
vpa	Vertical position absolute

To construct a database entry for a new printer, see details about the structure of the **terminfo** database in the **terminfo(4)** manual page.

Once you've made the new entry, you need to compile it into the database using the **tic(1)** program. Just enter the following command:

```
# tic file-name ↵
```

file-name is the name of the file containing the **terminfo** entry you have crafted for the new printer. The LP print service gains much efficiency by caching information from the **terminfo** database. If you add or delete **terminfo** entries, or change the values that govern pitch settings, page width and length, or character sets, you should stop and restart the LP print service so it can read the new information.

How to write an interface program

If you have an interface program that you used with the LP service before DG/UX Release 5.4, it should still work with the LP print service. Note, though, that several **-o** options have been “standardized,” and will be passed to every interface program. These may interfere with similarly named options used by your interface.

If you have a printer that is not supported by simply adding an entry to the **terminfo** database, or if you have printing needs that are not supported by the standard interface program, you can furnish your own interface program. It is a good idea to start with the standard interface program, and change it to fit, rather than starting from scratch. You can find a copy of it under the name **/var/spool/lp/model/standard**.

What does an interface program do?

Any interface program is responsible for doing the following tasks:

- Initializing the printer port, if necessary. The generic interface program uses the **stty** command to do this.
- Initializing the physical printer. The generic interface program uses the **terminfo** database and the **TERM** shell variable to get the control sequences to do this.
- Printing a banner page, if necessary.
- Printing the correct number of copies of the request content.

An interface program is not responsible for opening the printer port. The LP print service opens the port, a process which includes calling a “dial-up” printer, if one is used to connect the printer. The printer port connection is given to the interface program as standard output, and the printer is identified as the “controlling terminal” for the interface program so that a “hang-up” of the port will cause a **SIGHUP** signal to be sent to the interface program.

A customized interface program must not terminate the connection to the printer or “uninitialize” the printer in any way.

How is an interface program used?

When the LP print service routes an output request to a printer, the interface program for the printer is invoked as follows:

```
/var/spool/lp/admins/lp/interfaces/P ID user title copies options f1 f2 ...
```

Arguments for the interface program are:

<i>P</i>	printer name
<i>id</i>	request ID returned by the lp(1) command
<i>user</i>	username of the user who made the request
<i>title</i>	optional title specified by the user
<i>copies</i>	number of copies requested by the user
<i>options</i>	blank-separated list of options specified by the user or set by the LP print service
<i>f1, f2, ...</i>	full pathnames of the files to be printed

When the interface program is invoked, its standard input comes from **/dev/null**, its standard output is directed to the printer port, and its standard error output is directed to a file that will be given to the user who submitted the print request.

The standard interface recognizes the following values in the blank-separated list in *options*.

nobanner	This option is used to skip the printing of a banner page; without it, a banner page is printed.
nofilebreak	This option is used to skip page breaks between separate data files; without it, a page break is made between each file in the content of a print request.

cpi=decimal-number1

lpi=decimal-number2

These options specify a format of *decimal-number1* columns per inch and *decimal-number2* lines per inch, respectively. The standard interface program extracts from the **terminfo** database the control sequences needed to initialize the printer to handle the character and line pitches.

The words **pica**, **elite**, and **compressed** are acceptable replacements for *decimal-number1* and are synonyms, respectively, for 10 columns per inch, 12 columns per inch, and as many columns per inch as possible.

length=decimal-number1

width=decimal-number2

These options specify the length and width, respectively, of the pages to be printed. The standard interface program extracts from the **terminfo** database the control sequences needed to initialize the printer to handle the page length and page width.

stty='stty-option-list'

The *stty-option-list* is applied after a default *stty-option-list* as a set of arguments to the **stty** command. The default list is used to establish a default port configuration; the additional list given to the interface program is used to change the configuration as needed.

lpd='argument-list' This option is used internally by the **lpsched** command; you can ignore it.

flist='file-list' This option is used internally by the **lpsched** command; you can ignore it.

The above options may be specified by the user when issuing a print request. Alternatively, they may be specified by the LP print service from defaults given by the administrator either for the printer (**cpi, lpi, length, width, stty**) or for the preprinted form used in the request (**cpi, lpi, length, width**).

Additional printer configuration information is passed to the interface program in the following shell variables:

TERM=printer-type

This shell variable specifies the type of printer. The value is used as a key for getting printer capability information from the **terminfo** database.

FILTER='pipeline' This shell variable specifies the filter to use to send the request content to the printer; the filter is given control of the printer.

CHARSET=character-set

This shell variable specifies the character set to be used when printing the content of a print request. The standard interface program extracts from the **terminfo** database the control sequences needed to select the character set.

A customized interface program should either ignore these options and shell variables or should recognize them and treat them in a consistent manner.

Customizing the interface program

Make sure that the custom interface program sets the proper **stty** modes (terminal characteristics such as baud rate and output options). The standard interface program does this, and you can follow suit. Look for the section that begins with the shell comment

```
## Initialize the printer port
```

Follow the code used in the standard interface program. It sets both the default modes and the adjusted modes given by either the LP print service or the user with a line such as the following:

```
stty mode options 0<&1
```

This command line takes the standard input for the **stty** command from the printer port. An example of an **stty** command line that sets the baud rate at 1200 and sets some of the option modes is shown below.

```
stty -parenb -parodd 1200 cs8 cread clocal ixon 0<&1
```

One printer port characteristic not set by the standard interface program is hardware flow control. The way that this is set will vary, depending on your computer hardware. The code for the standard interface program suggests where this and other printer port characteristics can be set. Look for the section that begins with the shell comment

```
# Here you may want to add other port initialization code.
```

Because different printers have different numbers of columns, make sure the header and trailer for your interface program correspond to your printer. The standard interface program prints a banner that fits on an 80-column page (except for the user's title, which may be longer). Look in the code for the standard interface program for the section that begins with the shell comment

```
## Print the banner page
```

The custom interface program should print all user related error messages on the standard output or on the standard error. The messages sent to the standard error will be mailed to the user; the messages printed on the standard output will end up on the printed page where they can be read by the user when he or she picks up the output.

When printing is complete, your interface program should exit with a code that shows the status of the print job. Exit codes are interpreted by the LP print service as shown in Table 9-9.

Table 9–9 Printer exit codes

Code	Meaning to the LP print service
0	The print request has been completed successfully. If a printer fault has occurred, it has been cleared.
1 to 127	A problem has been encountered in printing this particular request (for example, too many non-printable characters, or the request exceeds the printer capabilities). The LP print service notifies the person who submitted the request that there was an error in printing it. This problem will not affect future print requests. If a printer fault had occurred, it has been cleared.
128	Reserved for internal use by the LP print service. Interface programs must not exit with this code.
129	A printer fault has been encountered in printing the request. This problem will affect future print requests. If the fault recovery for the printer directs the LP print service to wait for the administrator to fix the problem, the LP print service will disable the printer. If the fault recovery is to continue printing, the LP print service will not disable the printer, but will try printing again in a few minutes.
greater than 129	These codes are reserved for internal use by the LP print service. Interface programs must not exit with codes in this range.

As the table shows, one way of alerting the administrator to a printer fault is to exit with a code of 129. Unfortunately, if the interface program exits, the LP print service has no choice but to reprint the request from the beginning when the fault has been cleared. Another way of getting an alert to the administrator (that does not require the entire request to be reprinted) is to have the interface program send a fault message to the LP print service but wait for the fault to clear. When the fault clears, the interface program can resume printing the user's file. When the printing is finished, the interface program can give a zero exit code just as if the fault had never occurred. An added advantage is that the interface program can detect when the fault is cleared automatically, so that the administrator doesn't have to enable the printer.

Fault messages can be sent to the LP print service using the **lp.tell** program. This is referenced using the **LPTELL** shell variable in the standard interface code. The program takes its standard input and sends it to the LP print service where it is put into the message that alerts the administrator to the printer fault. If its standard input is empty, **lp.tell** does not initiate an alert. Examine the standard

interface code immediately after these comments for an example of how the **lp.tell** (LPTELL environment variable) program is used:

```
# Here's where we set up the $LPTELL program to capture
# fault messages.

# Here's where we print the file.
```

If the special exit code 129 or the **lp.tell** program is used, there is no longer a need for the interface program to disable the printer itself. Your interface program can disable the printer directly, but doing so will override the fault alerting mechanism. Alerts are sent only if the LP print service detects the printer has faulted, and the special exit code and the **lp.tell** program are its main detection tools.

If the LP print service has to interrupt the printing of a file at any time, it will terminate the interface program with a signal `TERM` (signal 15; see **kill(1)** and **signal(2)**). If the interface program dies from receipt of any other signal, the LP print service assumes that future print requests won't be affected, and continues to use the printer. The LP print service notifies the person who submitted the request that the request has not been finished successfully.

When the interface is first invoked, the signals `HUP`, `INT`, `QUIT`, and `PIPE` (trap numbers 1, 2, 3, and 13) are ignored. The standard interface changes this so that these signals are trapped at appropriate times. The standard interface interprets receipt of these signals as warnings that the printer has a problem; when it receives one, it issues a fault alert.

How to write a filter

A filter is used by the LP print service each time it has to print a type of file that isn't acceptable by a printer. A filter can be as simple or as complex as needed; there are only a few external requirements:

- The filter should get the content of a user's file from its standard input and send the converted file to the standard output.
- A **slow** filter can send messages about errors in the file to standard error. A **fast** filter should not, as described below. Error messages from a **slow** filter are collected and sent to the user who submitted the file for printing.
- If a **slow** filter dies because of receiving a signal, the print request is stopped and the user who submitted the request is notified. Likewise, if a **slow** filter exits with a non-zero exit code, the print request is stopped and the user is notified. The exit codes from **fast** filters are treated differently, as described below.

- A filter should not depend on other files that normally would not be accessible to a regular user; if a filter fails when run directly by a user, it will fail when run by the LP print service.

“Providing Filters,” earlier in this chapter, describes how to add a filter to the LP print service.

If you want your filter to detect printer faults, you must also fulfill the following requirements:

- If possible, the filter should wait for a fault to be cleared before exiting. Additionally, it should continue printing at the top of the page where printing stopped after the fault clears. If the administrator does not want this contingency followed, the LP print service will stop the filter before alerting the administrator.
- It should send printer fault messages to its standard error as soon as the fault is recognized. It does not have to exit, but can wait as described above.
- It should not send messages about errors in the file to standard error. These should be included in the standard output stream, where they can be read by the user.
- It should exit with a zero exit code if the user’s file is finished (even if errors in the file have prevented it from being printed correctly).
- It should exit with a non-zero exit code only if a printer fault has prevented it from finishing a file.
- When added to the filter table, it must be added as a **fast** filter. (See “Defining a filter” in this chapter for details.)

Translating character sets

To print a file that uses a character set different from the one used by your printer, use the **iconv** command to convert the file’s character set to one that is compatible with the printer’s. The **iconv** command uses the database, **/usr/lib/kbd/iconv_data**, which contains a list of code sets that can be converted. The code set being converted must be listed in this file. See the **iconv(1)** manual page for more information on this command and the **iconv_data** file.

For example, suppose you wanted to print a file that was created on an AOS/VS system using 8 bit DGI characters on a printer that supports IBM character sets 437 and 850. You can convert both of these files with standard System V Release 4 printer filters using **iconv** as the filter. Follow these steps to translate the files:

1. Create a filter with the input defined as “dgi” and output type of “88591”. The only supported conversion tables for DGI are to and from 88591. So, you must first map all DGI characters to ISO 8859.1 characters first and then map again for the required device

character set. In the filter, you specify the command **iconv** as the program, the from (**-f**) value as **dgi**, and the to (**-t**) value as **88591**. These values, are taken from the **iconv_data** file. The from value is column 1 in the file. The to value is column 2.

The entries for input, output, and content types are not specific indicators for the filter **iconv**. However, the values must match between the filters and the printers. The specific conversion is specified on the filter command line, so the entry, “**dgi**”, is the descriptive reference for the input type to the filter.

Here is the specific example for a filter called “**dgi_to_88591**”:

```
Input types: dgi
Output types: 88591
Printer types: any
Printers: any
Filter type: slow
Command: /usr/bin/iconv -f dgi -t 88591 -m b
```

2. Define a second filter to convert from the ISO 8859.1 character set to the IBM code page 850. Note that all the DGI characters map to the ISO 8859.1 character set, but not all are supported in the IBM 850 character set.

Define the input type for the filter as **88591** and the output type as **PC850**. **iconv** is again specified as the program with the from (**-f**) value as **88591** and the to (**-t**) value as **PC850**.

Here is the specific example for a filter called **88591_to_PC850**:

```
Input types: 88591
Output types: PC850
Printer types: any
Printers: any
Filter type: slow
Command: /usr/bin/iconv -f 88591 -t PC850 -m b
```

3. Define a printer of type “**ibmgraphics**.” The Content type is **PC850**, which matches the output type of the filter, **88591_to_PC850**. Below is the specific example for the printer called **ibm_850**. On the **ibmgraphics** printer, **character_set_1** is code page 437, and **character_set_2** is code page 850.

printer ibm_850 is idle. enabled since Mon Jul 13 16:56:13 EDT 1992. available.

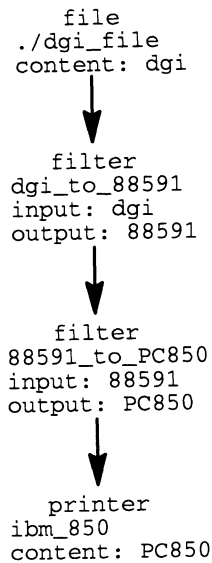
Form mounted:
Content types: PC850
Printer types: ibmgraphics
Description: direct connect ibm proprinter
Connection: direct
Interface: /usr/lib/lp/model/standard
On fault: mail to transou once
After fault: continue
Users allowed:
 (all)
Forms allowed:
 (none)
Banner not required
Character sets:
 character_set_1
 character_set_2
Default pitch: 10 CPI 6 LPI
Default page size: 80 wide 66 long
Default port settings:

To print a file that contains dgi characters on an IBM ProPrinter which supports the code page 850, you would enter the following lp command.

lp -d ibm_850 -T dgi -S character_set_2 /dgi_file ↵

In this command line -d selects the printer ibm_850, -T specifies the content type of the file as dgi, and -S selects character set 2 (code page 850) for the IBM printer.

The entire conversion of file follows this path:

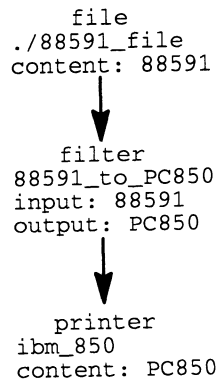


As another example, to print a file containing characters from the ISO 8859.1 character set on an IBM ProPrinter supporting the code page 850, you would enter the following lp command.

```
lp -d ibm_850 -T 88591 -S character_set_2 /88591_file ↵
```

In this command line, **-d** selects the printer `ibm_850`, **-T** specifies the content type of the file is `88591`, and **-S** selects character set 2 (code page 850) for the IBM printer.

The file conversion follows this path:



Using the man pages for `iconv(1)`, `lp(1)`, `lpadmin(1)`, and these examples, you should be able to define the necessary conversion filters for any of the characters sets supported in `/usr/lib/kbd` and the map file `/usr/lib/kbd/iconv_data`.

Managing the LP subsystem on a trusted DG/UX system

IMPORTANT: Data General offers B1 and C2 Trusted DG/UX systems. Both are optional and must be purchased in addition to the DG/UX system. This section applies only if you have purchased the B1 Trusted DG/UX system.

This section describes how to administer the line printer subsystem on a Trusted DG/UX system. The Trusted DG/UX system provides a trusted version of the LP subsystem, which meets the printout labeling requirement appropriate for the class of trust and features set for B1 level trusted systems. Minimally, the trusted LP subsystem labels printouts with the sensitivity label of the data.

The trusted LP subsystem features provide audit reports and MAC checking. These features are self configuring. When auditing is configured, **lpsched** automatically generates audit reports. When MAC is configured, **lpsched** does the appropriate MAC checking.

This section covers only the information necessary to administer the LP subsystem in a trusted environment. For information on how to configure printers on a trusted system, see Chapters 3, 4, 5, and 6 in this manual. For details on managing the Trusted DG/UX system and the differences between the DG/UX system and the Trusted DG/UX system, see *Managing Security on the Trusted DG/UX™ System*. For details about auditing, see *Managing Auditing on the Trusted DG/UX™ System*.

WARNING: If you overloaded this release on a previous security release, you must add your printer definitions again. If you overloaded this release on a regular DG/UX system, you must modify the print queues to include a MAC range. Do this by performing the modify operation in the the Device -> Printer -> Devices menu of sysadm. You will be prompted for the low and high values of the MAC range. Users cannot print to those queues until you modify them.

WARNING: If a system had remote printers added to the system before the Trusted DG/UX system was installed, these printers should be deleted before any lp-related activities are undertaken.

Labeling output

This section describes how the Trusted DG/UX system uses banner pages, headers, and footers to label printer output.

Banner pages

All printouts have front and back banner pages with the clearance label clearly marked. Figures 9–2 and 9–3 show sample front and back banner pages.

IMPORTANT: As system administrator, you can specify that no banner pages or only a front banner page be printed.

The banner pages include a label that is the Least Upper Bound (LUB) of the labels of all files included in the print request. The LUB is the minimum label that dominates all of the labels of the files in the print request. It is the highest hierarchy in the set of labels and contains all the categories.

If a page header/footer override is in effect, the LUB computation for the label on the banner page includes the clearance label of the **lp** process that submitted the request. (See “Page headers and footers” later in this section for information on header/footer override.) For example, if user **bob** logged in with clearance label of **ACCRED_HI** and prints two files (with no override) with labels of **IMPLEMENTATION_LO** and **ACCRED_LO**, the banner pages will show labels of **ACCRED_LO**. The individual pages will have headers and footers with **ACCRED_LO** or **IMPLEMENTATION_LO**, depending on the file. If **bob** prints the files using one of the override options, the banner page label will be **ACCRED_HI** (including his LP process label in the LUB computation).

If the LP subsystem cannot get an alias for the label, the print request is rejected unless the requestor has appropriate privilege.

If a print request is canceled or held during printing, the request is stopped. The back banner page prints before the next request starts. When a held print request is restarted, the front banner page is reprinted. Figures 9–2 and 9–3 are sample front and back banner pages, respectively.

```
#####  
farah  
User: monty!farah  
Job ID: ps1-250  
Printer: ps1  
Date: Fri Nov 11 08:32:20 EST 1992  
#####  
Printing the following files:  
    /var/spool/lp/tmp/D250-1  
#####  
#####  
This output must be protected as  
  
SYSTEM_HI  
  
unless manually reviewed and downgraded  
Bannerid: 1698941684  
#####  
#####  
#####
```

Figure 9-2 Sample front banner page

```
#####  
farah  
User: monty!farah  
Job ID: ps1-250  
Printer: ps1  
Date: Fri Nov 11 08:32:20 EST 1992  
#####  
  
END OF PRINTOUT  
  
#####  
#####  
This output must be protected as  
  
SYSTEM_HI  
  
unless manually reviewed and downgraded  
Bannerid: 1698941684  
#####  
#####  
#####
```

Figure 9-3 Sample back banner page

Protection against banner page spoofing

On the sample banner pages shown in Figures 9-2 and 9-3, the last item is titled `Bannerid` followed by a number. This number is randomly generated by the LP subsystem. The purpose of this number is to help prevent spoofing of the banner pages. That is, to

prevent a user from printing a fraudulent back burst page. The banner identification number is the same on both the front and back banner pages.

Page headers and footers

The Trusted DG/UX system adds a header and footer to each page that consists of the sensitivity label alias for the file being printed. The **lp** command has options that let you override the labels; however, using these options is an auditable event. The **-o nomachheader** option eliminates the header and footer label. The **-o machheader=<string>** option replaces the clearance label alias text with the given string. These options let you provide labels appropriate to each page of a document or eliminate the labels. A user might want to eliminate the labeling, for example, if the wordprocessing application being used provides for labeling in the header and footer.

For PostScript files, the page header and footer are overlaid on the page. They overwrite whatever other text or graphics are within the header and footer areas. A maximum of 10 lines each are reserved for the page header and footer. An additional 3 lines are used to separate the header and footer from the user's text and to provide a margin when printing ASCII files (input type of **simple**). Therefore, for a one line label, the total space used is 8 lines per page. For a 10-line label, the total space used is 26 lines per page. The print request for a non-PostScript (**simple**) input file is rejected if

$$\text{header length} + \text{footer length} + N > \text{page length}$$

where:

N = minimum number of lines permitted on a page
(currently 5 lines)

For ASCII (**simple**) input files, 3 lines are used in each of the header and footer, even if the **-o nomachheader** option is used. If a label is too long for the 10-line maximum, it is truncated to 10 lines and treated as an override.

If you are piping the output of another command that performs page breaks, be sure to remember the space used by these labels. For example, the **pr** command assumes 66 lines per page. To get correct page breaks for a file with a one line clearance label, use

```
# pr -l58 file | lp ↵
```

IMPORTANT: If you pipe a file into **lp**, the labels on the printout will reflect the process' level, which may be different from the file's level.

The clearance label that appears on the printout is computed when the labeling filter is run and not necessarily at the time of the request or the time the file is actually printed. The label printed is that of the file or files specified in the **lp** command, or the label of the **lp** command (usually the user's current clearance label) if input is piped to **lp**. For requests with multiple files, the page header and footer may change for each file.

For print requests with multiple files, the banner pages have a label consisting of the highest hierarchical level and the union of all the categories (that is, the LUB) for all files printed. Page labels will have the label from the current file.

Access control

On a B1 system, each print request has a label. The label for the request is the LUB label of all files included in the request and the LP process making the request. When a user issues the **lpstat** command, he sees only his own print requests that are dominated by his current process clearance label, unless he has appropriate privilege.

In a B1 system, printers have clearance ranges. Only requests with labels within the printer's range are accepted for that destination. When a user issues the **lpstat** command, he sees only printers with ranges such that the low end of the printer's range is dominated by the user's current clearance label.

The **-M** option has been added to the **lpstat** command to allow the user to get MAC information about print requests and printers.

Examples:

```
# lpstat -o -M ↵
```

```
lp-52      proto      2526   Mar  5 17:11 ACR_LO filtered
lp-53      proto       18    Mar  5 17:11 ACR_LO filtered
lp-54      proto       18    Mar  5 17:11 IMPL_HI filtered
```

```
# lpstat -p -M ↵
```

```
printer lp disabled since Thu Mar  5 15:22:58 EST 1992.
available.
```

```
unknown reason
```

```
MAC Range: IMPL_LO --> IMPL_HI
```

```
printer ps disabled since Thu Mar  5 10:30:53 EST 1992.
available.
```

```
unknown reason
```

```
MAC Range: ACR_LO --> ACR_LO
```

If the printer's high end range is not dominated by the user's current clearance label, the high end range is not displayed in the **lpstat** output.

Example:

```
# lpstat -p -M ↵

printer lp disabled since Thu Mar  5 15:22:58 EST 1992.
available.
    unknown reason
    MAC Range: IMPL_LO -->
printer ps disabled since Thu Mar  5 10:30:53 EST 1992.
available.
    unknown reason
    MAC Range: ACR_LO --> ACR_LO
```

Unlike standard DG/UX systems, a user cannot issue **enable** and **disable** commands to printers unless the user has appropriate privilege.

Auditing the LP subsystem

This section provides a brief overview of what **lpsched** audits. The LP subsystem security features are self configuring. If auditing is configured, **lpsched** automatically generates audit reports. Similarly, if MAC is configured, **lpsched** does the appropriate MAC checking.

IMPORTANT: If you overloaded the security release on a previous security release, you must add your printer definitions again.

lpsched C2 system auditing support

In the C2 system configuration, **lpsched** generates two types of audit reports. One audit report is associated with **lp** requests, and the other is associated with printers. **lpsched** generates printer audit records for the following events:

lp_add	Printer has been added.
lp_delete	Printer has been deleted.
lp_enable	Printer is enabled to print.
lp_disable	Printer is disable from printing.
lp_accept	Printer is set to accept requests.

For **lp** requests, **lpsched** generates audit records for the following events:

lp_request	Submit a request.
lp_hold	Hold a request from printing.
lp_resume	Resume printing a request.
lp_immediate	Print a request immediately.
lp_change	Modify an existing request.
lp_move	Move requests from one printer to another.
lp_cancel	Cancel a request.
lp_print	Send request to local printer.
lp_remote	Scheduler sends a request to a remote printer.
lp_nobanner	Print a request with no banner.

lpsched B1 system support

In the B1 system configuration, **lpsched** includes the MAC label of the client process when generating an LP request audit record. The MAC range of the printer is included when generating a printer audit record.

lpsched generates audit records for all events listed in the C2 system configuration and the following event:

lp_override	Print a request without headers and footers or with special headers and footers.
--------------------	--

Use the **-o nomacheader** or **-o macheader=** option with the **lp** command to make use of **lp_override**.

lpsched B1 system MAC label support

In the B1 system configuration each print queue has an associated MAC range, and each request has an LUB and a Greatest Lower Bound (GLB) label determined by the files to be printed.

An LP request can print only if the GLB and LUB labels fall within the MAC range for the print queue. In the case of a local printer, the GLB and LUB of the files must also fall within the MAC range of the physical device, such as **/dev/lp**.

A user can act on a request (for example, change or cancel) only if he owns the request and his MAC label dominates the LUB of the request.

If the print queues exist on a system before the B1 system features are installed, those queues must be modified to include a MAC range. Users are unable to print to those queues until the modification is performed.

End of Chapter

A Troubleshooting

This appendix provides simple troubleshooting tips for some of the most frequent printing problems. It also explains two systematic methods of troubleshooting complicated problems. The first method examines problems that can occur in the LP scheduler. The second examines the flow of data (the Data Flow Isolation technique (DFI)). This appendix also includes helpful information about the following:

- **terminfo** and forms
- Streams and printer devices
- MiLAN FastPort print servers
- TermServers
- Print filters
- Remote printing
- Printing with BSD

Quick troubleshooting tips

Here are a few suggestions of what to do if you are having difficulty getting a printer to work.

No Output (Nothing is Printed)

The printer is sitting idle; nothing happens. First, check the documentation that came with the printer to see if there is a self-test feature you can invoke; make sure the printer is working before continuing.

There are three possible explanations when you don't receive any output.

Is the Printer Connected to the Computer?

The type of connection between a computer and a printer may vary. See your printer and computer hardware documentation.

Is the Printer Enabled?

The printer must be enabled in two ways: first, the printer must be turned on and ready to receive data from the computer. Second, the LP print service must be ready to use the printer. If you receive error messages when setting up your printer, follow the "fixes" suggested in the messages. When the printer is set up, issue the commands

```
# accept printer-name ↵  
# enable printer-name ↵
```

where *printer-name* is the name you assigned to the printer for the LP print service. Now submit a sample file for printing:

```
# lp -d printer-name file-name ↵
```

Is the Baud Rate Correct?

If the baud rate (the rate at which data is transmitted) is not the same for both the computer and the printer, sometimes nothing will print (see below).

Illegible Output

The printer tries printing, but the output is not what you expected; it certainly isn't readable. There are four possible explanations for this situation:

Is the Baud Rate Correct?

Usually, when the baud rate of the computer doesn't match that of the printer, you'll get some output but it will not look at all like what you submitted for printing. Random characters will appear, with an unusual mixture of special characters and unlikely spacing.

Read the documentation that came with the printer to find out what its baud rate is. It should probably be set at 9600 baud for optimum performance, but that doesn't matter for now. If it isn't set to 9600 baud, you can have the LP print service use the correct baud rate (by default it uses 9600). If the printer is connected via a parallel port, the baud rate is irrelevant.

To set a different baud rate for the LP print service, enter the following command:

```
# lpadm -p printer-name -o stty=baud-rate ↵
```

Now submit a sample file for printing (explained earlier in this section).

Is the Parity Setting Correct?

Some printers use a “parity bit” to ensure that the data received for printing has not been garbled in transmission. The parity bit can be encoded in several ways; the computer and the printer must agree on which one to use. If they do not agree, some characters either will not be printed or will be replaced by other characters. Generally, though, the output will look approximately correct, with the spacing of “words” typical for your document and many letters in their correct place.

Check the documentation for the printer to see what the printer expects. The LP print service will not set the parity bit by default. You can change this, however, by entering one of the following commands:

```
# lpadmin -p printer-name -o stty=oddp ↵
# lpadmin -p printer-name -o stty=evenp ↵
# lpadmin -p printer-name -o stty=-parity ↵
```

The first command sets odd parity generation, the second sets even parity. The last command sets the default, no parity.

If you are also setting a baud rate other than 9600, you may combine the baud rate setting with the parity settings, as in the sample command below.

```
# lpadmin -p printer-name -o "stty='evenp 1200'" ↵
```

Are Tabs Set Correctly?

If the printer doesn’t expect to receive tab characters, the output may contain the complete content of the file, but the text may appear in a chaotic looking format, jammed up against the right margin (see below).

Is the Printer Type Correct?

See below under “Legible Printing but Wrong Spacing.”

Are the Default Port Settings (stty) Correct?

If you are printing binary data, the default port settings should include **-opost** which prevents any output processing by the streams device driver. Alternately, you can include **opost** in your **lp** command line: **lp file -o -opost**.

Legible Printing, but Wrong Spacing

The output contains all of the expected text and may be readable, but the text appears in an undesirable format: double spaced, with

no left margin, run together, or zigzagging down the page. These problems can be fixed by adjusting the printer settings (if possible) or by having the LP print service use settings that match those of the printer. The rest of this section provides details about solving each of these types of problems.

Double Spaced

Either the printer's tab settings are wrong or the printer is adding a linefeed after each carriage return. (The LP print service has a carriage return added to each linefeed, so the combination causes two linefeeds.) You can have the LP print service not send tabs or not add a carriage return by using the **stty -tabs** option or the **-onlcr** option, respectively.)

```
# lpadmin -p printer-name -o stty=-tabs ↵  
# lpadmin -p printer-name -o stty=-onlcr ↵
```

No Left Margin/Runs Together/Jammed Up

The printer's tab settings aren't correct; they should be set every 8 spaces. You can have the LP print service not send tabs by using the **-tabs** option.

```
# lpadmin -p printer-name -o stty=-tabs ↵
```

Zigzags Down the Page

The **stty onlcr** option is not set. This is set by default, but you may have cleared it accidentally.

```
# lpadmin -p printer-name -o stty=onlcr ↵
```

A Combination of Problems

If you need to use several of these options to take care of multiple problems, you can combine them in one list, as shown in the sample command below. Include any baud rate or parity settings, too.

```
# lpadmin -p printer-name -o "stty='-onlcr -tabs 2400'" ↵
```

Wrong Characters Printed

If the wrong printer type was selected when you set up the printer with the LP print service, the wrong "control characters" can be sent to the printer. The results are unpredictable and may cause

output to disappear or to be illegible, making it look like a problem described above. Another result may be that the wrong control characters cause the printer to set the wrong character set or font.

If you don't know which printer type to specify, try the following to examine the available printer types. First, if you think the printer type has a certain name, try the following command.

```
# tput -T printer-type longname ↵
```

The output of this command will appear on your terminal: a short description of the printer identified by the *printer-type*. Try the names you think might be right until you find one that identifies your printer.

If you don't know what names to try, you can examine the **terminfo** directory to see what names are available. Warning: there are probably many names in that directory. Enter the following command to examine the directory.

```
# ls -CR /usr/share/lib/terminfo/* | more ↵
```

Pick names from the list that match one word or number identifying your printer. Try each of the names in the other command above.

When you have the name of a printer type you think is correct, set it in the LP print service by entering the following command:

```
# lpadmin -p printer-name -T printer-type ↵
```

Dial Out Failures

The LP print service uses UUCP to handle dial-out printers. If a dialing failure occurs and you are receiving printer fault alerts, the LP print service reports the same error reported by UUCP for similar problems. (If you haven't arranged to receive fault alerts, they are mailed, by default, to the user **lp**.) See *Managing Modems and UUCP on the DG/UX™ System* for a list of UUCP error messages and explanations.

Idle Printers

There are several reasons why you may find a printer idle and enabled but with print requests still queued for it:

- The print requests need to be filtered. Slow filters run one at a time to avoid overloading the system. Until a print request has been filtered (if it needs slow filtering), it will not print. Use the following command to see if the first waiting request is being filtered.

```
# lpstat -o -l ↵
```

- The printer has a fault. After a fault has been detected, printing resumes automatically, but not immediately. The LP print service waits about five minutes before trying again, and continues trying until a request is printed successfully. You can force a retry immediately by enabling the printer as follows:

```
# enable printer-name ↵
```

- A dial-out printer is busy or doesn't answer, or all dial-out ports are busy. As with automatic continuation after a fault, the LP print service waits five minutes before trying to reach a dial-out printer again. If the dial-out printer can't be reached for an hour or two (depending on the reason), the LP print service finally alerts you to a possible problem. You can force a retry immediately by enabling the printer as follows:

```
# enable printer-name ↵
```

Networking Problems

You may encounter several types of problems while trying to get files printed over a network: (1) requests being sent to remote printers may back up in the local queue; (2) requests sent to remote printers may be backed up in the remote queue; or (3) a user may receive contradictory messages about whether a remote printer has accepted a print request. The rest of this section describes each of these situations and suggests how to resolve them.

Jobs Backing Up in the Local Queue

There are a lot of jobs backing up in the local queue for a remote printer. There are the following possible explanations:

- The remote system is down or the network between the local and remote systems is down. To resolve this problem, run the **reject** command for all the remote printers on your system, as follows:

```
# reject printer-name ↵
```

This will stop new requests for those printers from being added to the queue. Once the system comes up again, and jobs start being taken from your queue, type **accept printer** to allow new jobs to be queued.

- The remote printer is disabled on the local system. The underlying DG/UX Release 5.4 network software was not set up properly. For details, see **lpssystem(1M)**.

- The **tcp** listen service does not exist for **lp**. Use the **sysadm** operation Device → Port → Port Service → List to check for the listen service. If the service does not exist, make sure that **/etc/saf/_pmtab** contains the listen service entries for **lp** and **lpr**. Examples of these entries exist in **/etc/saf/_pmtab.proto**.
- The **tcp** listen service is disabled. Enter the **sacadm -l** command on the server to check the status of the port monitor. If the **tcp** status is not “ENABLED,” enter the following commands to restart the port monitor:

```
# sacadm -k -p tcp ↵
# sacadm -s -p tcp ↵
```

Wait for **sacadm -l** to show the monitor is enabled before trying to print.

Jobs Backing Up in the Remote Queue

The remote printer has been disabled.

Conflicting Acceptance/Rejection Messages

A user enters a print request and is notified that the system has accepted it. The job is sent to a remote system and the user receives mail that the job has been rejected.

This may be happening for one of two reasons:

- the local host may be accepting requests while the remote host is rejecting them, or
- the printer’s definition on the local host may not match its definition on the remote host.

The definitions of job components (such as filters, character sets, print wheels, and forms) must be identical on both the local and the remote systems if local users are to be able to access remote printers.

Troubleshooting printer problems in Ipsched

When the above tips do not solve your printer problem, try the following procedures:

1. Verify the physical connection and printer setup.
2. Verify the type of printer selected in **sysadm** is compatible with the physical printer.

3. Verify the interface script is compatible with the printer.
4. Verify all filters used are compatible with the document type and printer.

Verifying the physical connection and printer setup

This section contains a technique for parallel printers and one for serial printers. If the printer is attached to a parallel port, use the **cat** command to send a document to the printer. The document must be a type that the printer supports. For a character printer that accepts simple (ASCII) text, you could issue the following command:

```
cat /etc/passwd > /dev/lp
```

For a PostScript printer, pipe **/etc/passwd** through the **postprint** filter with the following command:

```
cat /etc/passwd | /usr/lib/postscript/postprint > /dev/lp
```

If the printer is able to produce a file that is reasonably close to what you expect, then the physical connection is probably a good one.

Serial connections require a bit more effort to verify. One technique, however, is to build a script that holds the port open and sends the document to the port. For a PostScript file, use the following script:

```
#!/bin/sh
sleep 32000 >/dev/tty21&
sleep_pid=$!

stty 9600 </dev/tty21
cat /etc/passwd | /usr/lib/lp/postscript/postprint > /dev/tty21
/bin/echo "\004\c" >/dev/tty21
kill -9 $sleep_pid
```

For an ASCII file, use the following script:

```
#!/bin/sh
sleep 32000 > /dev/tty21&
sleep_pid=$!

stty 9600 </dev/tty21
cat /etc/passwd > /dev/tty21
kill -9 $sleep_pid
```

Verifying the printer queue type is compatible with the printer emulation

The printer queue type has the same relation to a printer that the **TERM** variable in the shell environment does to a terminal. The

printer queue type identifies the **terminfo** database used by the **lp** interface script and the **lp** filters to initialize various features and modes on the printer.

To see what the current printer queue type is for a printer named **ps2**, for example, issue the **lpstat** command as follows:

```
$ lpstat -p ps2 -l
```

```
printer ps2 is idle. enabled since Wed Oct 9 16:22:20 EDT 1991.
available.
```

```
Content types: PS
Printer types: PS-b
Description: PS2
Connection: direct
Interface: /usr/lib/lp/model/standard
After fault: continue
Users allowed:
    (all)
Forms allowed:
    (none)
Banner not required
Character sets:
    ascii
    iso-88591
    iso-88592
    iso-88597
Default pitch:
Default page size:
Default port settings: clocal
```

In the above example, the printer queue type is **PS-b**.

To ensure that the printer queue type (**terminfo** entry) matches the printer emulation, test the printer queue type by sending a printer initialization sequence to the printer. To send an initialization sequence, use the **tput** command as follows:

```
$ tput -Ttype capname parameter
```

where:

-Ttype	Finds information for terminal named in the terminfo database.
capname	Specifies an attribute from the terminfo database.
parameter	Specifies an argument to be used in the string indicated by capname.

To test the initialization string of an hplaserjetII, issue **tput** as follows:

```
tput -ThplaserjetII init > /dev/lp
```

To test the initialization string of a hplaserjetIII, issue **tput** as follows:

```
tput -ThplaserjetIII init > /dev/lp
```

This command finds the initialization string in the **terminfo** database and sends it to the printer connected to **/dev/lp**. To analyze the strings sent to the printer, do the following:

```
tput -T hplaserjetIII init > filename  
od -cb filename |pg
```

The example above sends the initialization string for hplaserjetIII printers to a file and displays the results page by page. After sending the initialization sequence to the printer, repeat the physical printer connection test in the previous section to verify that the printer has been initialized properly.

Verifying the lp interface script is compatible with the printer connection method

The **lp** subsystem interface scripts initialize the printer using the **terminfo** database. These scripts print the banner page, set printer attributes, such as character set and lines per inch, and print each file in the request. For connections to the network and connections via **remshlp**, the interface script also ships the data to the remote system.

The **lp** subsystem requires different interface scripts for different printer connection types. Table A-1 lists and describes the available interface scripts.

Table A-1 Lp subsystem interface scripts

Interface script	Description
standard	Used for printers connected locally to /dev/lp or to a /dev/tty line.
remshlp_a	Used for remote AOS/VS II systems.

Continued

Table A-1 Lp subsystem interface scripts

Interface script	Description
remshlp_bsd	Used for remote BSD system.
remshlp	Used for remote DG/UX systems prior to Release 5.4.
fastport	Used for printers connected to the parallel or serial printer port of a MiLAN FastPort print server.
termprinter_3com3.3	Used for CS/2000 and 2100 Series TermServers.
termprinter_tcp	Used for CS/3000 and 3100 Series TermServers and HP LaserJet 4 printers with network adapter cards.
termprinter_telnet	Used for printers connected to a specific telnet port.
dg_annex	Used for both the Annex and MicroAnnex Termservers.
datability	Used for the VCP300 and 1000 TCP Series Termservers.

To determine the current interface script for a given printer, use the **lpstat** command. For a printer named **ps_doc**, you would enter the following:

```
$ lpstat -p ps_doc -l | grep Interface
```

```
Interface: /usr/lib/lp/model/fastport
```

The above example shows the interface script is **fastport**.

Verifying remote system information for remshlp and network connections

Printer queues that use the **remshlp** interface script or a network interface script require additional connection information. For this class of printer queues, you must supply a remote hostname and port number or remote printer name.

To see this connection information, use the following command:

```
$ lpstat -p printer -l | grep Connection
```

For printers connected to the network using the **fastport** or **termprinter** interface script, the connection data appears as follows:

```
Connection: fastport2-ps%2000
```

This indicates that the FastPort server is using the hostname **fastport2-pr**. The port number is 2000.

For printers that use the **remshlp** interface script, the data appears as follows:

```
Connections: ps2@jester
```

This indicates that the remote host is **jester** and the remote printer name is **ps2**.

When you have displayed the connection information, verify that the remote host or printer server can be accessed on the network by using the **ping** command as follows:

```
$ ping hostname
```

If the host can be accessed, output from the **ping** is:

```
hostname is alive
```

Troubleshooting using data flow isolation techniques

DFI attempts to isolate a problem to a module. The module may be a piece of hardware or a piece of software. DFI isolates most module failures and most configuration problems. Once the failing module has been isolated, the problem can range from a broken printer to supplying the wrong option to a filter. For hardware problems, see the manual for your particular printer. For software problems, see *Managing the DG/UX™ System*.

DFI is time-consuming, but effective in isolating a problem. You must, however, have a detailed understanding of the **lp** subsystem. The major steps used in the DFI technique are:

- Determine the data flow path.
- Locate possible break points in the data flow path.
- Trap the data at the break points.
- Analyze the data to isolate the problem to a particular module.

This section presents two examples that illustrate the DFI technique. The first involves a single machine using local printing. The second involves two dissimilar machines using remote printing.

The main objective in this technique is to determine the complete route the data takes to get to the printer and examine the data along the route. For example, a PostScript printer has a content type PS, which means that the printer can accept only files that are in postscript format. When a simple document (not PostScript) is sent to this printer, the data passes through a series of filters:

```
datafile -> postprint -> lp.crn1 -> lp.cat -> /dev/lp -> printer
```

You can trap and examine the output of any of the filters. At any point, you can send the trapped data file directly to the printer. Thus, you can test the data flow incrementally from the data file to the port and isolate a problem to a module or device.

Example 1

This example uses the printer definition below, which is located in **/etc/lp/printers/PS/configuration**.

```
Banner: on:Always
Content types: PS
Device: /dev/lp
Fault: wait
Interface: /usr/lib/lp/model/standard
Printer type: PS-b
Stty: clocal
Modules: keep
```

Suppose you want to print a simple (ASCII) document on this printer. Because the content type of the printer is **PS**, the simple text document must be converted to a PostScript document by a slow filter then routed to the fast filters. To see how filters work, examine the **/etc/filter.table** file. The slow filter, **postprint**, accepts a simple input file and produces a **PS** output file. The **lp** subsystem selects this filter and routes the data through it. To capture the filtered data, follow these steps.

1. Disable the PostScript printer. This allows you to trap the output from the slow filter. Submit a simple ASCII file (**testfile**) to the disabled printer (**psfake**) with the following command:

```
# lp -dpsfake /usr/barker/testfile ↵
request id is PS-169 (1 file)
```

2. Examine the printer status. Note the status shows the document has been filtered, but is not on the printer.

```
# lpstat -o ↵
PS-169      barker      36 Aug 31 08:43 filtered
```

3. Locate the filtered file in `/var/spool/lp/tmp/system`.

```
# ls /var/spool/lp/tmp/csc603 ↵
148-0      169      169-0      F169-csc603-1
```

F169-csc603-1 is the filtered file, indicated by the "F." This is the simple data file after it has been converted to PostScript format. The data flow up to here is:

```
tmp/testfile -> postprint -> F169-csc603-1
```

The original file (`testfile`) contents are as follows:

```
line 1
line 2
line 3
LINE 4
LINE 5
```

The initial lines of the filtered file (F169-csc603-1) contents are as follows:

```
%!PS-Adobe-2.0
%%Version: 3.15
%%DocumentFonts: (atend)
%%Pages: (atend)
%%EndComments
/vmstat save def%
% Copyright (C) Data General Corporation, 1984 - 1991
% All Rights Reserved.
% Licensed Material-Property of Data General Corporation.
% This software is made available solely pursuant to the
% terms of a DGC license agreement which governs its use.
%
%      <@(#) postprint.ps,v 5.1.1.2>

%ident "@(#)lp:filter/postscript/postscript/postprint.ps
1.1"
%
% Version 3.15 prologue for ASCII files.
%
```

The initial lines of the filtered file show that the file successfully converted to a PostScript format suitable for printing to a PostScript printer.

4. Use the `cat` command to send this file to the printer.

```
cat /var/spool/lp/tmp/csc603/F169-csc603-1 > /dev/lp ↵
```

If the file does not print, examine the physical connection and physical setup of the printer. The **cat** command is the lowest level tool available through the operating system for trouble shooting and can be considered a very simple fast filter. For a serial printer, use the following to send a PostScript document directly to the printer:

```
sleep 3200 > /dev/tty20 & ↵
stty clocal 9600 -opost < /dev/tty20 ↵
cat file /dev/tty20 ↵
```

If the file does not print, examine the physical connection, or physical printer setup.

5. To find the filters that the interface file routes the data through and the environment passed to the interface file, add the diagnostic lines below to the interface file in directory **/etc/lp/interfaces**. Insert the following lines at the beginning of the file:

```
echo > /tmp/PS.INFO      # Clear the file
echo $1 $2 $3 $4 $5 $6 >> /tmp/PS.INFO #Get arguments passed
env >> /tmp/PS.INFO      # Get the environment
```

Insert the following line just before the **exit** statement at the end of the file:

```
echo "${FILTER}" >> /tmp/PS.INFO      # Get the filters
```

6. Print the test file.

```
# lp -dPS testfile
```

7. Examine the file **/tmp/PS.INFO**.

```
# cat /tmp/PS.INFO
```

```

PS-171 csc603!barker 1 stty='clocal'
/var/spool/lp/temp/F171-csc603-1
DISPLAY=:0.0
FCEDIT=vi
FILTER=/usr/lib/lp/bin/lp.cat 0
HOME=/usr/disk1/barker
LANG=C
LOGNAME=barker
MAIL=/usr/mail/barker
NLSPATH=/usr/lib/nls/msg/%L/%N:/etc/nls/msg/%L/%N
PATH=/sbin:/usr/sbin:/usr/bin:/etc
PS1=#
PWD=/usr/disk1/barker
SHELL=/usr/bin/ksh
SPOOLER_KEY=1128163897
TERM=PS-b
TERMCAP=vs|xterm|vs100|xterm-65|xterm terminal emulator (X window
system) :
:AL=\E[%dL:DC=\E[%dP:DL=\E[%dM:DO=\E[%dB:IC=\E[%d@:MT:UP=\E[%dA:
:al=\E[L:am:
:bs:cd=\E[J:ce=\E[K:cl=\E[H\E[2J:cm=\E[%i%d;%dH:co#80:
:cs=\E[%i%d;%dr:ct=\E[3k:          :dc=\E[
P:dl=\E[M:do=^J:
:ei=\E[4l:ho=\E[H:im=\E[4h:is=\E[r\E[m\E[2J\E[H\E[?7h\E[?1;3;4;6l
:
:k1=\EOP:k2=\EOQ:k3=\EOR:k4=\EOS:kb=^H:kd=\EOB:ke=\E[?1l\E>:
:kl=\EOD:kn#4:kr=\EOC:ks=\E[?1h\E=:ku=\EOA:
:le=^H:li#24:md=\E[1m:me=\E[m:mi:mr=\E[7m:ms:nd=\E[C:p
t:          :rs=\E[r\E<\E[m\E[2J\E[H\E[?7h\E[?1;3;4;6l:
:se=\E[m:sf=\n:so=\E[7m:sr=\EM:
:te=\E[2J\E[?47l\E8:ti=\E7\E[?47h:
:ue=\E[m:up=\E[A:us=\E[4m:xn:
TZ=EST5EDT
USER=barker
VISUAL=vi
WINDOWID=92274706
XAPPLRESDIR=/usr/disk1/barker/app-defaults/
_=/usr/bin/su
folder=/usr/disk1/barker/mail/files
/usr/spool/lp/bin/lp.crn1 | (/usr/lib/lp/bin/lp.cat 0; code=$?;
echo '

```

The first line in the output above contains the arguments passed to the script file. The last line shows the filters that the data are piped through. The data between the first and last line represent the environment passed to the interface file.

The last line is the one you use to determine the data flow path. The environment and argument list are useful in identifying the

cause of a problem after the problem has been isolated to a module.
The last line in the example is:

```
/usr/spool/lp/bin/lp.crn1 | (/usr/lib/lp/bin/lp.cat 0; code=$?; echo '
```

The complete data flow path through the fast filters is as follows:

```
datafile -> lp.crn1 -> lp.cat -> /dev/lp -> printer
```

Thus, a simple document sent to a printer with content type PS travels the following route:

```
simpledoc -> postprint -> lp.crn1 -> lp.cat -> /dev/lp -> printer
```

On the other hand, if the document is of type PS, you can determine the data flow without changing the diagnostic statements. For example, you send a document of type PS (**ps.doc**) to a printer (**ps2**) as follows:

```
# lp -d ps2 ps.doc
```

The last line in the file **/tmp/PS.INFO** is:

```
/usr/spool/lp/bin/lp.crn1 | (/usr/lib/lp/lp.cat; code=$?; echo '
```

The data flow path through fast filters for a file of type PS is virtually the same as that for simple files. The slow filter **postprint**, however, is not used because the document is already in PostScript format.

In summary, to use the DFI technique, you must first determine the complete data flow path. You then select isolation points to look at the data. You isolate the module containing the problem. Finally, you determine why the problem occurred in the module.

Example 2

The DFI technique can be extended to any number of modules or interconnected machines. Consider two AViONs and a PC. The PC is connected to AV1 via a third party communications package. The AViONs are connect via TCP/IP. The printer is attached to AV2 and the PC wants to print to it. The overall data flow is:

```
PC -> AV1 -> AV2 -> printer
```

The DFI technique is the same. You determine the data flow path and isolate the problem to either a hardware or software module. The following example illustrates the technique.

The hardware configuration includes: a PC connected to an AViiON (AV1) via a third party package: a second AViiON (AV2) attached to AV1 via TCP/IP; a serial printer called **lpt** attached to AV2. The PC wants to print to **lpt**. The PC package emulates BSD printing. The data flow path is:

PC_doc -> BSD emulation package -> AV1 -> AV2.

Use the following command on AV1 to display the printer setup for printer csc602:

```
# cat /etc/lp/printers/csc602/configuration ↵
```

```
Content types: simple
Printer type: printer-80
Remote: csc602!lpt
```

The **/etc/lp/Systems** entry on AV1 for the PC is:

```
PC:x:-:BSD:-:10:10:-:-:
```

Use the following command to display the printer setup on AV2:

```
# cat /etc/lp/printers/lpt/configuration ↵
```

```
Banner: on:Always
Content types: simple
Device: /tmp/printer1
Fault: continue
Interface: /usr/lib/lp/model/standard
Printer type: printer-80
Modules: keep
```

The **/etc/lp/System** entry on AV2 for AV1 is:

```
csc603:x:-:s5:-:10:10:-:-:
```

We have no way of determining the printer setup on the third party PC. To use this troubleshooting procedure, follow these steps.

1. Determine if the PC document is getting to AV1 by disabling the printer on AV1 and submitting a print job from the PC.

```
# disable csc602 ↵
```

2. Issue the appropriate print request from the PC.
3. Check the status of the print request.


```
# lpstat -o ↵
PC-175                PC!barker            36   Aug 27 11:24
```

4. Find the job PC-175.

```
# cd /var/spool/lp ↵
# find . -print |grep 175 ↵
./requests/PC/175-0 ↵
./tmp/PC/175-0 ↵
./tmp/PC/175-1
```

The **./tmp/PC/175-0** file is a specification file that contains a variety of information. The **./tmp/PC/175-1** is the data file. Both files can be examined at this point to isolate problems.

5. Use the **cat** command to examine the **tmp** files named above.

```
# cd /var/spool/lp ↵
# cat 175-0 ↵
C 1
D PC
F /var/spool/lp/tmp/csc603/175-1
P 20
t simple
U barker
s 0x1000
```

```
# cat 175-1 ↵
line 1
line 2
line 3
LINE 4
LINE 5
```

You can trap and examine the document by disabling the printer on **AV1**. If this fails, the problem is in the communication between the PC and **AV1** or a failure of the PC software.

To trap the document on **AV2**, disable the printer on **AV2** and enable the queue on **AV1**. Submit a job, using the technique illustrated above. As shown with these two examples, the techniques are expandable. Considering **AV1**, **AV2** and the PC as modules, you can trap and examine the data in each machine. When we have isolated the problem to a machine. You use the same technique to isolate the problem to a module as in example one.

The steps are the same for a single machine printing problem or a multi-machine, multi-vendor printing problem.

- Determine the data flow path.
- Select break points in the path.
- Trap the data at the break points.
- Isolate the module.
- Identify the problem in the module.

Troubleshooting specific areas and programs

This section provides helpful information about the following:

- **terminfo** and forms
- Streams and printer devices
- MiLAN FastPort devices
- TermServers
- Print filters
- Remote printing
- Printing with BSD

Terminfo and forms

This section applies only to printers without an lpi (lines per inch) parameter, but with micro-stepping capabilities. *Micro-stepping* means that the printer has a stepping motor that requires a given number of pulses to move the paper up an inch. This type of vertical spacing is common in dot matrix printers. For these printers, the LP subsystem must calculate the number of micro-steps for various lpi settings.

The Epson2500-hi dot matrix printer is an example of a printer that uses micro-stepping. If, for example, you want 6 lines per inch, the LP subsystem must find out how many steps per inch the printer supports and calculate how many steps will provide 6 lines per inch. The Epson2500-hi has 180 steps per inch. So if you divide 180 by 6, you get the number of steps necessary before the next line is printed. The number of steps is 30.

To understand how the LP subsystem sets Lines Per Inch (lpi) or Characters Per Inch (cpi) parameters, you must understand the language of **terminfo** and how it is used. The **terminfo** database is written in the **curses** language. It is like an interpreter type of language, with the syntax in the style of ALGO-68 or FORTH. Like FORTH, **curses'** concept of a stack is that of variables that are last in and first out.

The **curses** language consists of operators that process variables on the stack. When **curses** processes the variables, it discards the variables and places only the result on the stack. For example, suppose you have a function called **foo** defined in the **terminfo** database as follows:

```
foo=%p1%p2%+%c
```

The function indicates that **foo** requires two arguments, adds them, and prints the result. Assume argument one is a 2 and argument two is a 4, **foo** is interpreted as follows:

%p1	Pushes argument one on the stack. The stack now has a 2 on it.
%p2	Pushes argument two on the stack. The stack now contains 4 and 2.
%+	Adds the two top values and pushes the sum on the stack. The stack discards the 4 and 2. The sum of 6 is pushed on the stack. Only the sum of 6 remains on the stack.
%c	Prints the value. The 6 is sent to standard output and the stack is now empty.

The **terminfo** language has several operators and a conditional (if-then-else-endif) construct. For details, see the **terminfo(4)** manual page.

The following example explains how to set the lines per inch parameter for the Epson 2500-hi printer to 6. The command **infocmp epson2500-hi** shows that the lpi capability is not available, but the orvi parameter and the cvr capability are present. These parameters are used for micro-spacing. They allow the LP subsystem to determine how many steps are necessary for a specific lines per inch setting.

The orvi parameter tells how many steps per inch the printer supports. The LP subsystem uses that information to determine how many steps are necessary to get the lines per inch you want. After the LP subsystem has determined the number of required micro-steps, it uses the cvr parameter to determine the ASCII codes sent to the printer to cause the printer to move the correct number of micro-steps.

The LP subsystem retrieves the orvi parameter from **curses** calls. This parameter, however, can be retrieved manually with either the **infocmp** or **tput** command. The command **Tput -Tepson2500-hi orvi** returns 180, which means that the Epson2500-hi supports 180 micro-steps per inch. To set the printer for 6 lines per inch, you

divide 180 by 6 to get the number of steps required to produce a 6 lines per inch document. The last step is to send the code to the printer to request 30 micro-steps per line.

The `cvr` parameter contains the instructions or program for sending the correct codes to the printer. The instructions can be seen by using either **infocmp epson2500-hi** or **tput -Tepson2500-hi cvr lod -cb**. The command **infocmp epson2500-hi** produces:

```
cvr=%?%p1%{0}%>%p1%{256}%<%&%t\E3%p1%c%;
```

The command **tput -Tepson2500-hi cvr lod -cb** produces:

```
0000000 % ? % p 1 % { 0 } % > % p 1 % {
          045 077 045 160 061 045 173 060 175 045 076 045 160 061 045 173

0000020 2 5 6 } % < % & % t 033 3 % p 1 %
          062 065 066 175 045 074 045 046 045 164 033 063 045 160 061 045

0000040 c % ;
          143 045 073
```

The two results are the same except the string shown by **infocmp** has a `\E`. The string shown by **tput** has a `033` in place of the `\E`. The `\E` sends an ASCII escape character, which is an octal `033`. Thus, the `033` and `\E` are actually the same.

The result of the **infocmp** command constructs the ASCII sequence that is sent to the printer. Translating the program to understandable pseudo code, we get the following:

Assuming 6 lines per inch:

```
orvi = 180 180 / 6 = 30
```

This means that the variable `p1 = 30`

```
if p1 > 0 and p1 < 256 then
print 033 063 30
end if
```

The real code translates as follows:

```
%?          if
%p1         push p1 on the stack. The stack = 30
%{0}       push a literal 0 on the stack. The stack = 0 30
%>         if the second value on the stack, 30, is greater
           than the first value, 0, pop both values and
           push a 1 (true) on the stack. The stack = 1.
```

```

%p1          push p1, 30, on the stack. The stack = 30 1.
%{256}      push the literal 256 on the stack. The stack =
            256 30 1.
%<          if the second value on the stack is less than the
            first, then push a 1 (true) on the stack.
            Otherwise, push a 0 (false). Pop both values off
            the stack. The stack = 1 1
%&          do a bitwise and of the two top values. Pop the
            two values and push the result of the and. In
            this case, the result of the and is 1 (true). The
            stack = 1.
%t          then. The value is popped off the stack.
033         print the ASCII escape 033.
3          print a 3 ASCII 063.
%p1         push p1 30 on the stack. The stack = 30
%c         print the top value on the stack and ASCII 30
            octal 036.
;          end of the if then statement.

```

The result is that the octal sequence 033 063 036 is sent to the printer. **Terminfo** entries that have an **lpi** parameter use this parameter rather than the **orvi** and **cvr** parameters discussed above. The **terminfo** entry for a **HplaserjetIII**, for example, has an **lpi** parameter. The **infocmp** command yields a more complicated result for **HplaserjetIII**. The results of the command **infocmp hplaserjetIII** are as follows:

```

lpi=?%p1%{1}%=%t\E&11D%e%p1%{2}%=%t\E&12D%e%p1%{3}%=%t\E&13D%e%p1%{4}%=%t\E&
14D%e%p1%{6}%=%t\E&16D%e%p1%{8}%=%t\E&18D%e%p1%{12}%=%t\E&112D%e%p1%{16}%=%t\
E&116D%e%p1%{24}%=%t\E&124D%e%p1%{48}%=%t\E&148D%; ,

```

The result from **tput -ThplaserjetIII | od -cb** is as follows:

```

0000000 % ? % p 1 % { 1 } % = % t 033 & 1
      045 077 045 160 061 045 173 061 175 045 075 045 164 033 046 154
0000020 1 D % e % p 1 % { 2 } % = % t 033
      061 104 045 145 045 160 061 045 173 062 175 045 075 045 164 033
0000040 & 1 2 D % e % p 1 % { 3 } % = %
      046 154 062 104 045 145 045 160 061 045 173 063 175 045 075 045
0000060 t 033 & 1 3 D % e % p 1 % { 4 } %
      164 033 046 154 063 104 045 145 045 160 061 045 173 064 175 045
0000100 = % t 033 & 1 4 D % e % p 1 % { 6
      075 045 164 033 046 154 064 104 045 145 045 160 061 045 173 066
0000120 } % = % t 033 & 1 6 D % e % p 1 %
      175 045 075 045 164 033 046 154 066 104 045 145 045 160 061 045
0000140 { 8 } % = % t 033 & 1 8 D % e % p
      173 070 175 045 075 045 164 033 046 154 070 104 045 145 045 160
0000160 1 % { 1 2 } % = % t 033 & 1 1 2 D
      061 045 173 061 062 175 045 075 045 164 033 046 154 061 062 104
0000200 % e % p 1 % { 1 6 } % = % t 033 &
      045 145 045 160 061 045 173 061 066 175 045 075 045 164 033 046
0000220 1 1 6 D % e % p 1 % { 2 4 } % =
      154 061 066 104 045 145 045 160 061 045 173 062 064 175 045 075
0000240 % t 033 & 1 2 4 D % e % p 1 % { 4
      045 164 033 046 154 062 064 104 045 145 045 160 061 045 173 064
0000260 8 } % = % t 033 & 1 4 8 D % ;
      070 175 045 075 045 164 033 046 154 064 070 104 045 073
0000276

```

The only difference in the results of the two commands is that **infocmp** displays a \E for the ASCII octal code 033, which is the same escape character.

The entire program is an if-then-else statement. The pseudo code translation is as follows:

```

if p1 = 1 then
    send the printer the ascii codes 033 046 154 061 104
else
if p1 = 2 then
    send the printer the ascii codes 033 046 154 062 104
else
if p1 = 3 then
    send the printer the ascii codes 033 046 154 063 104
else
if p1 = 4 then
    send the printer the ascii codes 033 046 154 064 104
else
if p1 = 6 then
    send the printer the ascii codes 033 046 154 066 104
else
if p1 = 8 then
    send the printer the ascii codes 033 046 154 070 104
else
if p1 = 12 then
    send the printer the ascii codes 033 046 154 061 062 104
else
if p1 = 16 then
    send the printer the ascii codes 033 046 154 061 066 104
else
if p1 = 24 then
    send the printer the ascii codes 033 046 154 062 064 104
else
if p1 = 48 then
    send the printer the ascii codes 033 046 154 064 070 104
end if

```

STREAMS and printer devices

A stream is a full duplex (bi-directional) processing and data transfer path between a STREAMS driver in the kernel space and a process in user space. A stream is constructed in the kernel by linking a stream head, zero or more modules, and a driver. The stream head is the end nearest the user process. It processes all system calls made by the user level process.

A module is a defined set of kernel-level routines and data structures used to process data, status, and control information. Data processing can involve changing the way data are represented; adding and deleting header and trailer information around the data; and placing data in or taking data out of packets. Status and control information includes signals and input/output control information. Each module is self-contained, communicating only with the component immediately before and immediately after it. Modules are not a required components in STREAMS.

A STREAMS driver may be a device driver that provides the services of an external I/O driver, commonly referred to as a pseudo-device driver. The driver typically handles data transfer between the kernel and the device and does little or no processing of data other than conversion between data structures used by the STREAMS mechanism and data structures understood by the device.

Modules can be used to perform intermediate processing of messages as the messages pass between the stream head and driver. This process of inserting one or more modules into a STREAM is called pushing a module on the stream. Modules are dynamically interconnected in a stream by a user process. No kernel programming, assembly, or link editing is required to create the interconnection. Figure A-1 shows an example of specific modules on a stream.

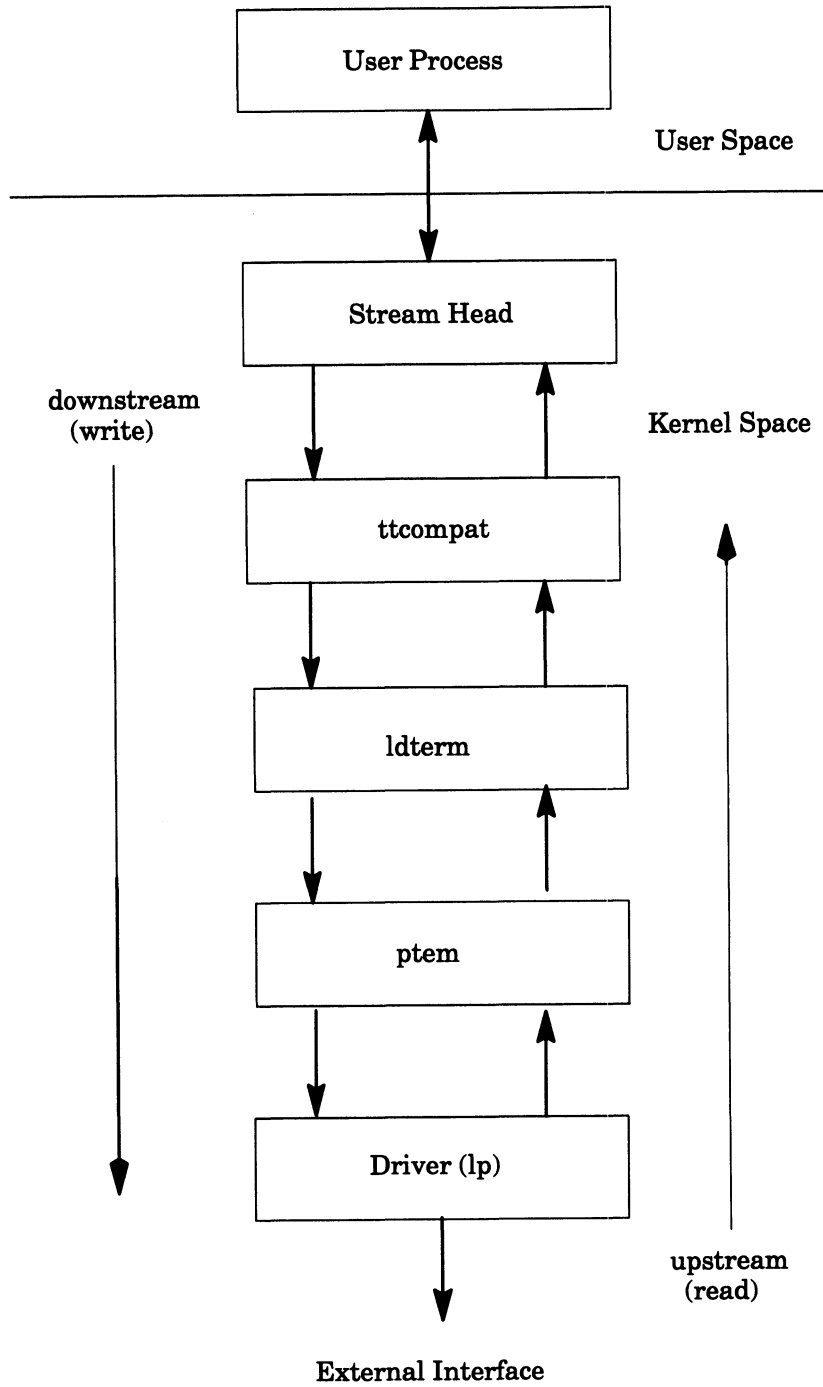


Figure A-1 Streams data transfer

Pushing modules on the /dev/lp device

Each serial line on a DG/UX system is a STREAMS device (ttyxx lines). The lp parallel port is also a STREAMS device. This allows

an application to push and pop modules onto **/dev/lp** and perform unique operations using the **ioctl** system call.

As a STREAMS device, **/dev/lp** provides the same interface as a tty device. This allows **stty** options to be passed to the STREAMS module. For example, a printer named **raw_lp** mapped to the device **/dev/lp** can be setup with no output processing as follows:

```
lpadmin -praw_lp -v/dev/lp -ostty=raw
```

You can then issue the **lp** command to print a bitmap graphics file to **/dev/lp** as follows:

```
lp -draw_lp arrow.pcl
```

The **lp** device is an example of how modules are pushed on a STREAMS device.

You can use the command **autopush** to print the list of modules automatically pushed on the **/dev/lp** device. First, use the command **ls -l** to provide the major and minor device number (9 and 0 respectively) for the device. Then supply these numbers as arguments to **autopush** as shown below:

```
ls -l /dev/lp ↵
crw-rw--rw-  1 root  root  9,  0  Aug 26  10:50  /dev/lp
autopush -g -M 9 -m 0
Major      Minor      Lastminor  Modules
  9         ALL        -          ptem ldterm ttcompat
```

Using the ttcompat, ldterm, and ptem streams modules

The modules **ttcompat**, **ldterm**, and **ptem** are typically pushed on the STREAM for printing devices.

The module **ttcompat** is the closest to the STREAM head. It takes older Version 7, 4BSD and XENIX **ioctl** calls, converts them into **ioctl** calls supported by the **termio** interface, and passes them downstream (see the manual page **termio(7)**).

The **ldterm** module is called the line discipline module. It performs many of the functions specified by the **stty** options. This module takes care of the post-processing on output, write, or down-stream side of the STREAM. The **stty** option **opost** enables post processing (**-opost** disables all post-processing) The output options enabled are:

OLCUC Map lower case to upper case on output

ONLCR	Map NL to CR on output
OCRNL	Map CR to NL on output
ONOCR	NO CR output at column 0
ONRLET	NL performs CR function
OFILL	Use fill characters for delay
OFDEL	Fill is DEL, else NULL
NLDLY	Select newline delays NL0 NL1
CRDLY	Select carriage–return delays or tab expansion CR0 CR1 CR2 CR3
TABDLY	Select horizontal tab delays TAB0 TAB1 TAB3 Expand tabs to spaces XTABS Expand tabs to spaces
BSDLY	Select backspace delays BS0 BS1
FFDLY	Select form feed delays FF0 FF1

Default settings are OPOST, ONLCR, and TAB3.

Post processing can cause problems when you are printing binary files. A binary file contains information that does not represent ASCII characters. For example, a binary file may contain information about downloading a font. An octal <011> might be a number that means a specific line number of font information. However, in ASCII characters, an octal <011> is a TAB. If OPOST is turned on along with TAB3, the <011> is expanded to a number of spaces, causing the <011> octal to be changed to an octal <040>. This can cause the font not to download properly. You should not use output processing for:

- Printing files that contain escape sequences to do specific functions on modern printers.
- Printers that have their own program language.
- Postscript printers that are printing bitmap graphic images.

The **ptem** module is the **STREAMS** pseudo-terminal emulator module. This module emulates a terminal. It handles all of the terminal type **IOCTL** commands and is pushed onto the slave side of a pseudo-terminal **STREAM**. This module translates **ioctl** commands into something that the hardware module below it can understand.

Troubleshooting parallel printers connected to **/dev/lp**

The notes in this section apply for most parallel printers connected locally to **/dev/lp**. When you cannot print using the scheduler and suspect a hardware problem, first, verify that **lp()** is configured into the kernel. Disable the printer from **lpsched** and remove the **STREAMS** modules so that **/dev/lp** can be accessed as a raw device. Follow the procedures below:

1. Verify that **lp()** is configured into the kernel.

```
# sysdef | grep lp\(\) ↵
lp()
#
```

2. Disable the printer.

```
# disable /dev/lp ↵
printer /dev/lp now disabled
#
```

3. Verify that nobody is using **/dev/lp**.

```
# fuser /dev/lp ↵
/dev/lp:
#
```

If the **fuser** command returns a PID number after **/dev/lp:** or if the command does not return to the shell prompt, then some process has **/dev/lp** exclusively opened. You must kill that process before continuing. Use the **ps** and **grep** commands to find the process as follows:

```
# fuser /dev/lp ↵
/dev/lp:      22602
# ps -ef | grep lp ↵
  root 21368 21363  7 15:38:38 ?  0:00 lpNet
  root 22602 21363  7 10:35:22 ?  0:00 /usr/lib/lpsched
  root 21363     1  7 15:38:38 ?  0:02 /usr/lib/lpsched
  root 22605 22603  7 10:35:54 p3 0:00 grep lp
```

PID 22602 is a child **lpsched** process that is holding **/dev/lp** open. If the **fuser** command hangs, interrupt the command with a **Ctrl-C**.

You will not be able to tell for sure what process has the port open. If there are child **lpsched** processes, such as in the above example, you can kill them. If the **fuser** command hangs again, use the **lpshut** command to bring **lpsched** down as follows:

```
# lpshut ↵
Print services stopped
#
```

4. Get the major and minor node numbers for **/dev/lp** using the **ls** command with the **l**, **i**, and **a** options.

```
# ls -lia /dev/lp
81 crw-rw-rw- 1 root root 9, 0 Jan 28 15:38 /dev/lp
#
```

The “9” and “0” are the major and minor node numbers respectively.

5. Check the current configuration of STREAMS modules using the **autopush** command with the **-g** option.

```
# autopush -g -M 9 -m 0 ↵
Major Minor Lastminor Modules
          9      0      -      ptem ldterm ttcompat
#
```

6. Remove the current STREAMS modules so that you can test the port in raw mode using the **-r** option.

```
# autopush -r -M 9 -m 0 ↵
```

7. Verify that the STREAMS modules have been removed using the **-g** option.

```
# autopush -g -M 9 -m 0 ↵
autopush: ERROR: Could not get autopush information
autopush: ERROR: Major/minor not configured for
autopush
#
```

8. Create a small text file such as **tfile**.

```
# cat tfile ↵
line 1
line 2
line 3
#
```

9. Use the **cat** command to send **tfile** to **/dev/lp**.

```
# cat tfile > /dev/lp ↵
#
```

To print a PostScript file, cat **tfile** as follows:

```
# cat tfile | /usr/lib/lp/postscript/postprint > /dev/lp ↵
#
```

The file should print and be readable, but if one of the following problems occurs,

- a. The file does not print or the result is unreadable. This usually means that you have a hardware problem. Do not continue.
 - b. The **cat** command hangs or you get the error `/dev/lp: cannot create`. This indicates a cable or printer problem.
 - c. The output is readable but is stair-stepped down the page. Continue with step 10.
10. Create a file, such as `str_mod`, continuing the configuration information as shown.

```
# cat str_mod ↵
9 0 0 ptem ldterm ttcompat
#
```

11. Reconfigure the list of modules using the **-f** option followed by the file name.

```
# autopush -f str_mod ↵
#
```

12. Check that the STREAMS modules are configured correctly. Use the **autopush** command with the **-g** option.

```
# autopush -g -M 9 -m 0 ↵
Major Minor Lastminor Modules
          9      0      -      ptem ldterm ttcompat
#
```

13. Use the **cat** command to send the file to `/dev/lp`. Use the same command as in step 9. The file should now print without stair-stepping.

14. If the scheduler was stopped in step 3, restart it.

```
# /usr/lib/lpsched ↵
Print services started
#
```

MiLAN FastPort device

The FastPort box has three local ports. Port 2000 is for the parallel printer and port 2001 is for the serial printer. Port 2002 is the diagnostic port. You can connect to this port using telnet with the following command:

```
telnet internet_address 2002
```

As an alternative, you could use the hostname associated with the above Internet address in the file `/etc/hosts`.

IMPORTANT: DG/UX System 5.4 Release 2.10 is the earliest revision of the DG/UX system that has built-in support for the FastPort.

When setting up a printer definition, you must use the FastPort's hostname, the correct port, and the `fastport` model file. This filter places all of its output in a temporary file before a connection is made to the proper FastPort port. This is different from the operation of a "normal" model file operation. For information on setting up FastPort printer definitions, see Chapter 6 in this manual.

Troubleshooting the FastPort

Troubleshooting printing problems always presents special challenges. Troubleshooting printing problems that are scattered about a network sometimes seems impossible. The FastPort, however, offers a number of built-in features that can make troubleshooting easier. Dialing-in to a customer's host and using telnet to connect to the FastPort box to troubleshoot a problem is basically the same as using telnet to connect to the FastPort box from a host on the network. The following sections discuss tools available for troubleshooting.

The four LEDs on the front of the FastPort box indicate the state of the FastPort. They are labeled POWER, ETHERNET, PARALLEL, and SERIAL. Each light blinks during operation. Their meanings are described in the list below:

POWER	Indicates the unit is operating and the status of the IP address. If the IP address is configured, the light blinks one time per second. When not configured, the light blinks 5 times a second.
ETHERNET	Indicates traffic on the Ethernet. This light blinks when packets are sent or received for the Ethernet address programmed in the nonvolatile random access memory (NOVRAM) board. It also blinks for broadcast packets. This light blinks twice a second when in diagnostic mode.

- PARALLEL** Indicates activity on the parallel port. This light blinks when data is sent out of the parallel port.
- SERIAL** Indicates activity on the serial port. This light blinks when data is sent or received on the serial port.

When starting up, the FastPort does a self test and tries to resolve its IP address. The FastPort can resolve its IP from RARP, or you can enter the address manually into the nonvolatile RAM. You can use the **ping** command to check whether or not the FastPort's Internet network node is running.

When the FastPort fails during the self test, the LEDs blink to indicate various failures. The self-test repeats every 20 seconds, causing the LEDs to display the error. Note that this procedure occurs only when the unit is in normal mode (both front panel switches are in the up position). Table A-2 shows how to interpret the problem.

Table A-2 FastPort self-test problems

Problem	POWER	ETHERNET	PARALLEL	SERIAL
CPU	Blinking	Blinking	Blinking	Blinking
SRAM	Blinking	Blinking	Blinking	Off
EPROM	Blinking	Blinking	Off	Blinking
Ethernet	Blinking	Blinking	Off	Off
UART	Blinking	Off	Blinking	Off
Parallel	Blinking	Off	Blinking	Blinking
NVRAM	Blinking	Off	Off	Blinking

FastPort diagnostic mode

The FastPort box is an intelligent device that has its own microprocessor, memory, and built-in operating systems. The operating system has a diagnostic mode that provides communication using one character commands. You can access the diagnostic mode through two different methods. The first method is to connect a serial terminal directly to the serial port on the FastPort box.

The terminal must be set-up in 8-bit mode and you must use a DB9 connector. This allows you to talk directly to the FastPort box using the terminal. You must then place the FastPort into direct-connect

diagnostic mode by setting the two switches on the front of the FastPort box in the down position and turning the power off and back on.

The second method to access the FastPort in diagnostic mode is by setting the left switch on the front of the FastPort box in the up position and the right switch in the down position. This allows users to print to the two printers off the FastPort, but also allows you to use the **telnet** command to connect to port 2002 of the box.

Both methods allow you to use the commands documented in the *FastPort 3000 and 3100 Network Print Server User's Guide*. In diagnostic mode, the FastPort accepts single character commands that must be followed by a carriage return to have the FastPort act on them. The most useful of these commands for troubleshooting are:

- | | |
|--------------------|--|
| D (display) | Displays the Ethernet address, IP address, parallel port status and serial port parameters. |
| H (help) | Displays a list of commands that are available in diagnostic mode. |
| R (reset) | Resets the FastPort box after ending a diagnostic session begun by using telnet to connect to port 2002. |
| E (enable) | Enable (or disables) password protection to restrict access to port 2002. |
| * | Sends a test page to a non-PostScript printer that accepts ASCII characters. |
| + | Sends a test page for a PostScript printer. |

The test page sent by the ***** and **+** commands appears as follows:

FastPort Model 3100 with High Speed DMA
Revision 3.11, Datecode 12/8 1992 15:23
Processor Speed = 10 MHz
Static Ram size = 256K bytes
Novram size = 128 bytes
Ethernet address 0 40 c8 2 3 1d
Ethernet options: Ignore Alignment Errors
IP Address: nnn.nnn.nn.nnn
Telnet password security: OFF

Netware options:

Auto-sense ethernet type between Ethernet II and 802.3

Console status monitoring configured as:

Off

Serial port: Baudrate Xon/Xoff Data bits Parity Stop bits
9600 Yes 8 none 1

Parallel port: mode test page setup (1/4 us) strobe (1/4 us)
Standard Ascii text 3

Parallel options: Return Parallel status

To leave the diagnostic mode, you need to do one of the following:

- When using the direct connect method, place the switches on the front of the FastPort box into the up position and turn the power off and then on.
- When using telnet to connect to port 2002, enter the **R** command to reset the FastPort. Jobs that are printing when you enter the **R** command will be lost. While in diagnostic mode, there doesn't seem to be a way to prevent the FastPort from accepting printing jobs.

The normal operating mode for the FastPort is to have the switches on the front of the box in the up position. In this mode, you cannot use telnet to connect to port 2002, but you can use telnet to connect to ports 2000 and 2001.

You can type text (with no echoing) and send it to the port you are logged into when you press Enter or New Line. This way, you can test any characters. The only way to break this connection is to turn off the power. This kills jobs currently printing on the FastPort.

IMPORTANT: You cannot type text if you are connected to a PostScript printer.

Normal mode is preferred for normal operations unless you want to allow restricted access to port 2002. Password protection to restrict access to port 2002 is available only in diagnostic mode.

FastPort switches

The following switch positions determine which Ethernet cable is being used with the box.

Table A-3 FastPort switch positions for Ethernet cables

Cable	Switch D1	Switch D2	Lights	
			coax	UTP
AUI-Thick Net (10Base5)	Up	Up	off	off
Thin Coax (10Base2)	Down	Up	on	off
UTP (10BaseT)	Up	Down	off	on

Potential problems

The following is a list of potential problems:

- During installation, assigning an Internet address that has already been assigned to another device can give unpredictable results. Be careful to assign unique addresses.
- During installation, if unusual problems occur or the FastPort does not seem to work the way it's documented, check the firmware revision of the box. We recommend units that are revision 3.2e or later. Use the **S** command while in diagnostic mode.
- While troubleshooting, notify users not to send jobs to printers that are attached to the FastPort box. These jobs can be lost when you change switch settings, turn off the power, or use the **R** command in diagnostic mode.

Termserver

DG/UX supports two methods of establishing the network connection between the AViiON host and the printer port. One method uses a VME-bus Terminal Controller, or VTC. The VTC is configured as a syac controller in the kernel, and is accessed through the `/dev/ttyxx` entries created for its 256 tty ports. The TCP/IP protocol is down-loaded onto the VTC; the host is not required to run TCP/IP. The other method uses a standard Ethernet controller, such as an **hken**, **dgen**, **cien**, or **inen**, or any controller that provides a host-based TCP/IP interface to the network.

Termserver printing using a VTC involves a setup in the DG/UX system that is similar to that of a local device. In addition to the

problems associated with local printing, the user must configure a means of establishing a network connection using TCP/IP between the host computer and a particular port of a termserver.

In TCP/IP, a network connection is established using a connect request to a particular IP address/tcp port number combination. For a network printer, the connection type can be either telnet or a raw tcp connection, depending on the capabilities of the host and the termserver.

VTC to termserver printing

The VTC appears as an asynchronous controller with 256 tty ports. The connection type used in **sysadm** when adding a VTC printer is **Local /dev/tty or /dev/lp**. This establishes the association between the print queue and a particular **/dev/ttyxx** file. The network connection is configured using a file called **/etc/tcload/vtc.addrs**. There, a particular tty is associated with a particular IP address/tcp port number.

The VTC allows a tty to be configured to accept an inbound connection to its IP address/tcp port, or to initiate an outbound connection to its associated IP address/tcp port. This function is determined by the Protocol field of a tty-specific entry in the **vtc.addrs** file. The protocol settings relevant for termserver printing are:

telnetin	Accepts inbound telnet connections.
tcpin	Accepts inbound raw tcp connections.
telnetout	Initiates outbound telnet connections to the printing device.
tcpout	Initiates outbound raw tcp connections to the printing device.

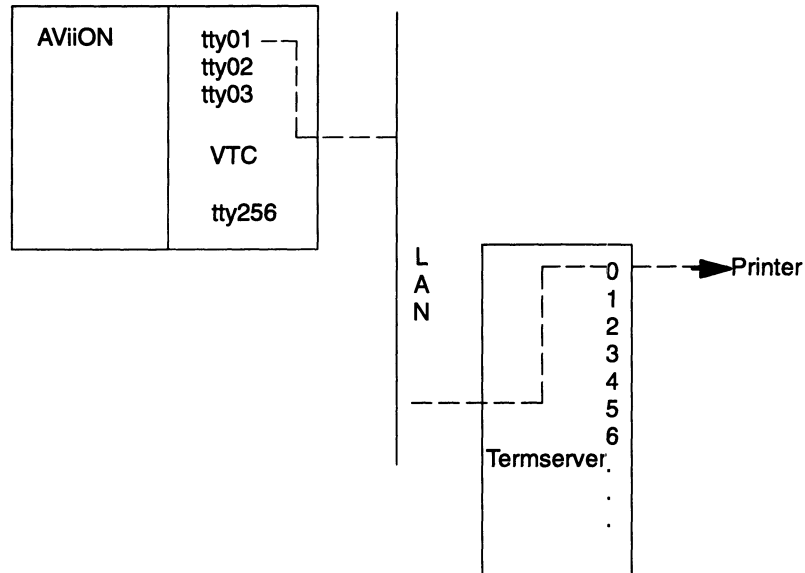


Figure A-2 VTC to termserver configuration

For example, if we have a network configured for tty01 to ACCEPT network connections from the printer attached to port 0 of the termserver, we would make one of the following entries in the file **/etc/vtc.addrs**:

/dev/tty01 128.222.72.252

or

/dev/tty01 128.222.72.252 @1000 tcpin

With the first entry, tty01 awaits a connection from a printer port using the TELNET protocol. The default tcp port number is 23 for TELNET, and is not specified. Similarly, the binary flag defaults to on, and does not need to be specified. For more information, see the manual page **vtc.addrs(4M)**.

With the second entry, tty01 awaits a raw tcp connection from the printer port at IP address 128.222.72.252/tcp port 1000.

With both entries, the printer port is configured to always connect to the IP address/tcp port on tty01. On a DG 2x00 Termserver, the port parameter that provides this function is called a Permanent Virtual Circuit (PVC). The command syntax that causes port 0 to always connect to tty01 for the first entry above is:

CS# setd i0 -term pvc = "128.222.72.252"

The command syntax for the second entry is:

CS# setd !0 -term pvc = "128.222.72.252 1000"

On a Datability termserver, the port parameter that provides this function is called a DEDICATED port. The command syntax to provide this functionality for port 1 of the termserver is:

Local> change port 1 dedicated 128.222.72.252

or

Local> change port 1 dedicated 128.222.72.252/1000

Local> change port 1 permanent enable

Local> change port 1 telnet disabled lpr enabled

The last command inhibits telnet negotiations from occurring while the connection is being established (that is, raw tcp mode).

IMPORTANT: In both of the above examples, tty01 will not accept the connect request until the print queue has been started, and a print request has been spooled to the tty. Also, while adding the printer in **sysadm**, specify the **stty** option **-hupcl** to prevent the network connection from being torn down between print requests.

As an alternative, you can set up a configuration to initiate an outbound connection to the printer port. The advantage of this method is that the connection is initiated only when a print job is submitted for the tty. That way, there are no unnecessary messages on the network. To establish an outbound connection, make one of the following entries in the file **/etc/vtc.adrds**:

/dev/tty01 128.222.72.252 telnetout

or

/dev/tty01 128.222.72.252 @2000 tcpout

The printer port can be configured in one of two ways. Either the printer port is assigned its own unique IP address and port number or the printer port uses the main IP address of the termserver, but is assigned its own unique tcp port number.

For a printer attached to a DG 2x00 Termserver, to assign an IP address to a printer attached to port 0, the command is as follows:

CS# setd !0 -tcpappl porm = 128.222.72.252

To allow you to use the main IP address assigned to the termserver, DG termserver can have a set of tcp port numbers assigned to each port. The tcp port numbers are determined by setting the **TNBasePort** parameter for telnet connections, or **TCPBasePort**

parameter for raw tcp connections. A printer port's tcp port number is equal to its physical port number plus the TNBasePort number for telnet, or it's physical port number plus the TCPBasePort for raw tcp connections. So, for the second example of an entry in the **vtc.addrs** file above, with the tcpport given, the setup on a DG 2x00 termserver would be:

```
CS # setd -tcpappl tcpbaseport = 2000
```

Now the printer attached to port 0 should handle an inbound connection request in raw tcp mode for the address of the Termserver/tcp port 2000.

Printing from an Ethernet controller to a termserver

The connection type configured in **sysadm** when a standard Ethernet controller is used for the AViiON's network connection is **Remote to networked printing device**. The IP address of the Termserver device is determined from the hostname you supply for Networked printer device name and the port number you supply to **sysadm**.

The hostname you supply must be found in the file **/etc/host**, and the port number must match the tcp port number associated with the printer's termserver port. The model file used for this type of printer will be either **termprinter**, **fastport**, or **datability**, depending on the type of termserver used.

When a print request is made for printers using this configuration, **lpsched** passes the network information along with the other information relevant to the request to the interface script for that printer.

The interface script captures the printer initialization sequences, the banner page, and file data in a temporary file. The script then establishes a telnet connection to the networked print server device, sends the temporary file to the device, and closes the connection. Figure A-3 shows this process.

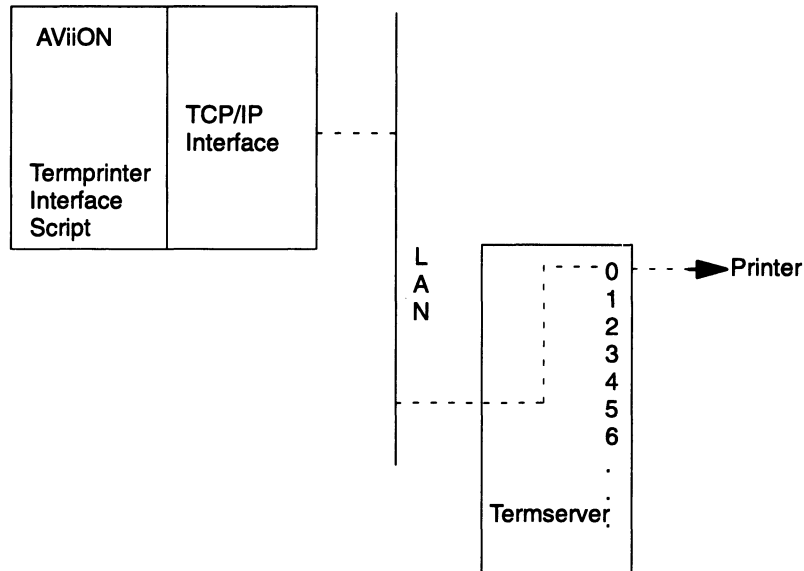


Figure A-3 Ethernet controller to termserver configuration

Termserver troubleshooting tips

When print jobs enter the print queue, but do not print, you should follow the procedures below.

The network connection should be formed after the print job enters the queue. If not, try the following steps:

1. Use the **ping** command to verify the IP address of the printer or the VTC port, whichever is set to accept incoming connections. If the response is *host is alive*, then at least someone is using that IP address. To verify it is on the correct network node, display the ARP table on the device from which you ran the **ping** command. The example below shows the command you issue from the DG/UX system, a 3Com termserver, and a datability termserver. You must have host based TCP/IP on the DG/UX system.

```
DG/UX:      arp -a
3Com:      sh -ip cache
DATABILITY: sh arp
```

If both are on the same network, the Ethernet address shown in the ARP table should match that of the VTC or Termserver port accepting the connection requests. If not, you should disable the IP

address on the device answering in error, or choose another IP address. If the **ping** command times out, verify the IP address being used for the printer, and check for an IP routing problem.

2. Telnet to the IP address/tcp port number assigned to the printer, using the appropriate command:

```
DG/UX:      telnet 128.222.72.252 1000
3Com:       telnet 128.222.72.252 1000
DATABILITY: c 128.222.72.252/1000
```

In each of the above examples, 1000 is the tcp port assigned for the printer. If the telnet request is satisfied, you receive a response indicating that you are connected. Then, anything you type on your keyboard is not be echoed back to your screen, but should be sent through the network, and appear on the printer. Some printers require a New Line, carriage return, or form feed character (^J, ^M, and ^L respectively) before they will print the data, so type those characters to see if the printer received the characters properly.

IMPORTANT: You may not be able to type text if you are connected to a PostScript printer. Some PostScript printers do not allow interactive sessions.

Several possible problems are listed below along with suggestions for resolving them.

- The printer prints garbage.
Check the baud rate, character length, and parity settings for the termserver port, and ensure they match those of the printer.
- The printer is waiting for a connection and does not print.
Check the cabling between the printer and the termserver port. Make sure that the pinouts are correct and the printer is connected to correct port.
- The VTC port is awaiting the connect request and there is a job in the queue waiting to print.
Your telnet request should connect. Then the print data, banner and all, should display on your screen. If there is something already connected to the VTC port, check the connection status of the termserver port. The example below shows the command for both 3Com and datability termserver.

```
3Com:       SHow AllSessions
DATABILITY: SHOW PORT x STATUS (x=physical port)
```

The printer's port should be the one connected to the VTC port. If not, look for another device on the network connecting to the VTC port in error. If the printer's port is connected to the VTC port, look

for a cabling or flow control problem between the printer and the termserver port.

- Print jobs enter the queue, appear to print, but do not come out of the printer.
Look for a duplicate of the printer's IP address and tcp port number combination on the same network.
- Garbage characters print at the beginning of a print job.
One end of the connection is using telnet protocol, the other using raw tcp. The telnet options negotiations from the telnet end are interpreted as data on the raw tcp end. Make both ends use raw tcp.
- Text documents print fine, but graphics documents print garbage.
 - a. If using telnet protocol, try using a raw tcp connection. Telnet manipulates the end-of-line characters in some modes, which produces garbage in a binary print job.
 - b. Try using the **-opost** stty option. This will prevent filters from interfering with the binary data.
 - c. If using a DG/3Com termserver, check the printer port's Default Parameter table for the CRPad, FFpad, TabPad, BSPad, and LFPad parameters. These should all be set to NONE. If set to a number, the termserver inserts that number of NULL characters following each FF, TAB, CR, BS, or LF.

Print filters

This section explains the **/etc/lp/filter.table** file. It defines the entries, presents a copy of the file, and presents a typical printing scenario, explaining how **lp sched** uses the filter table.

Defining filter table entries

The list below presents field definitions for each entry in the **filters.table.proto** file:

printer type	The list of printer types into which this filter can convert files.
printer names	The list of printers on which this filter can run. The filter uses the DG/UX system defined name for the printer. This entry is almost always set to any.
input to filter	The list of file content types that this filter can process. Most filters handle only one file type. LP print services uses these names to match a filter with a file type.

filter name	The filter name that appears in the sysadm filter section for this entry.
output from filter	The list of file types that this filter can produce as output. For each input type, the filter will produce one output type. In practice, there is normally only one input and one output type per entry.
filter type	<p>Either a slow or fast filter. Fast filters incur little overhead in preparing a file for printing and must have access to the printer when they run.</p> <p>Slow filters incur a lot of overhead in preparing a file for printing and don't require access to a printer. The LP print service runs slow filters in the background. This allows files that don't need slow filtering to move ahead.</p>
pathname to filter	The pathname to the filter that this entry needs to run. Switches that the filter needs can be also passed here. This entry is also known as command.
options	Options the filter uses. See postprint (1) manual page for examples.

Sample filter.table file

In the sample **filter.table** file below, the encircled numbers above the fields (columns) of the first entry (row) match the numbers of the definitions above. An encircled uppercase letter appears before each entry (row). The fast filters are entries **A** through **F** and the slow filters are **G** through **O**. The encircled numbers and uppercase letters are not a part of the **filter.table.proto** file. They appear only to aid in this discussion.

```

# .
# Copyright (C) Data General Corporation, 1984 - 1991
# All Rights Reserved.
# Licensed Material-Property of Data General Corporation.
# This software is made available solely pursuant to the
# terms of a DGC license agreement which governs its use.
#
# <@(#) filter.table.i,v 5.1.1.12>

#ident "@(#)lp:filter/postscript/filtdesc/filter.table.i 1.5"
#
# Factory installed filter table
#
  ① ② ③ ④ ⑤ ⑥ ⑦ ⑧
A: PS:any:postdown:postio:PS:fast:/usr/lib/lp/postscript/postio -L /dev/null:
B: PS-b:any:postdown:postio_b:PS-b:fast:/usr/lib/lp/bin/lp.cat 0:
C: PSR,PS-r:any:postrev:postio_r:PSR,PS-r:fast:/usr/lib/lp/postscript/postio -L
  /dev/null:
D: PS-br:any:postrev:postio_br:PS-br:fast:/usr/lib/lp/bin/lp.cat 0:
E: any:any:simple:epsonfx:epsonfx:fast:/usr/lib/lp/bin/epsonfx -a0:TERM * =
  -T*,MODES bold = -b,MODES color\=\(\(.*\)\) = -c\1,MODES double_height =
  -d,MODES htabs\=\(\(.*\)\) = -h\1,MODES justify\=\(\(.*\)\) = -j\1,MODES
  left\=\(\(.*\)\) = -l\1,MODES proportional = -p,MODES right\=\(\(.*\)\) =
  -r\1,MODE S shadow = -s,MODES tray\=\(\(.*\)\) = -t\1,MODES
  vtabs\=\(\(.*\)\) = -v\1,MODES draft = -qdraft,MODES normal =
  -qnormal,MODES nlq = -qnlq,MODES graphic_chars = -C,MODES
  bottom\=\(\(.*\)\) = -S\1,MODES timeout\=\(\(.*\)\) = -a\1
F: any:any:simple:proprinter:proprinter:fast:/usr/lib/lp/bin/proprinter -a0:TERM
  * = -T*,MODES bold = -b,MODES color\=\(\(.*\)\) = -c\1,MODES
  htabs\=\(\(.*\)\) = -h\1,MODES left\=\(\(.*\)\) = -l\1,MODES proportional
  = -p,MODES right\=\(\(.*\)\) = -r\1,MODES shadow = -s,MODES
  tray\=\(\(.*\)\) = -t\1,MODES vtabs\=\(\(.*\)\) = -v\1,MODES draft =
  -qdraft,MODES normal = -qnormal,MODES nlq = -qnlq,MODES char_set_2 =
  -C,MODES bottom\=\(\(.*\)\) = -S\1,MODES timeout\=\(\(.*\)\) = -a\1

G: PS,PSR,PS-r,PS-b,PS-br:any:postscript:download:postdown:slow:/usr/lib/lp/f
  postscript/download:PRINTER * = -p*
H: any:any:daisy:postdaisy:postscript:slow:/usr/lib/lp/postscript/postdaisy:
  PAGES * = -o*,MODES group = -n2,MODES group\=\(\([2-9]\)\) =
  -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES
  x\=\(\(-*\[\0-9]\*\)\) = -x\1,MODES y\=\(\(-*\[\0-9]\*\)\) =
  -y\1,MODES magnify\=\(\([\0-9]*\)\) = -m\1,MODES
  copies\=\(\([0-9][0-9]*\)\) = -c\1,MODES font\= * = -f*,MODES psize\=
  * = -s*,MODES pointsize\=\(\([\0-9]*\)\) = -s\1

```

①	②	③	④	⑤	⑥	⑦	⑧
<p>Ⓜ:any:any:troff:dpost:postscript:slow:/usr/lib/lp/postscript/dpost:PAGES * = -o*,LENGTH * = -l*,MODES group = -n2,MODES group=\([2-9]\) = -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES x=\(\(-*\[0-9]\)\) = -x\1,MODES y=\(\(-*\[0-9]\)\) = -y\1,MODES magnify=\([0-9]\) = -m\1,MODES copies=\([0-9][0-9]\) = -c\1,MODES font\= * = -f*,MODES psize\= * = -s*,MODES pointsize=\([0-9]\) = -s\1</p>							
<p>Ⓜ:any:any:dmd:postdmd:postscript:slow:/usr/lib/lp/postscript/postdmd:PAGES * = -o*,LENGTH * = -l*,MODES group = -n2,MODES group=\([2-9]\) = -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES x=\(\(-*\[0-9]\)\) = -x\1,MODES y=\(\(-*\[0-9]\)\) = -y\1,MODES magnify=\([0-9]\) = -m\1,MODES copies=\([0-9][0-9]\) = -c\1</p>							
<p>Ⓜ:any:any:matrix:postmd:postscript:slow:/usr/lib/lp/postscript/postmd:PAGES * = -o*,LENGTH * = -l*,MODES group = -n2,MODES group=\([2-9]\) = -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES x=\(\(-*\[0-9]\)\) = -x\1,MODES y=\(\(-*\[0-9]\)\) = -y\1,MODES magnify=\([0-9]\) = -m\1,MODES dimen=\([1-9]*x*[1-9]\) = -d\1,MODES interval=\(\(-*[0-9]/*-*[0-9]/*-*[0-9]\)\) = -i\1,MODES copies=\([0-9][0-9]\) = -c\1</p>							
<p>Ⓜ:any:any:simple:postprint:postscript:slow:/usr/lib/lp/postscript/postprint: PAGES * = -o*,LENGTH * = -l*,CHARSET * = -S*,CHARSET [0-9][0-9]* = -Scs&,CHARSET iso-\([0-9]\) = -Sps\1,MODES group = -n2,MODES group=\([2-9]\) = -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES x=\(\(-*\[0-9]\)\) = -x\1,MODES y=\(\(-*\[0-9]\)\) = -y\1,MODES magnify=\([0-9]\) = -m\1,MODES copies=\([0-9][0-9]\) = -c\1,MODES font\= * = -f*,MODES psize\= * = -s*,MODES pointsize=\([0-9]\) = -s\1</p>							
<p>Ⓜ:any:any:postdown:postreverse:postrev:slow:/usr/lib/lp/postscript/postreverse: PAGES * = -o*</p>							
<p>Ⓜ:any:any:tek4014:posttek:postscript:slow:/usr/lib/lp/postscript/posttek:PAGES * = -o*,LENGTH * = -l*,MODES group = -n2,MODES group=\([2-9]\) = -n\1,MODES portrait = -pp,MODES landscape = -pl,MODES x=\(\(-*[0-9][0-9]\)\) = -x\1,MODES y=\(\(-*[0-9][0-9]\)\) = -y\1,MODES magnify=\([0-9][0-9]\) = -m\1,MODES copies=\([0-9][0-9]\) = -c\1,MODES font\= * = -f*,MODES psize\= * = -s*,MODES pointsize=\([0-9][0-9]\) = -s\1</p>							
<p>Ⓜ:any:any:simple:pageprint:simple:slow:/usr/lib/lp/bin/pageprint:TERM * = -T*,CPI * = -c*,LPI * = -l*,LENGTH * = -L*,WIDTH * = -W*,PAGES * = -o*, MODES ff = -f</p>							

When lpsched receives a request from lp, lpsched first determines if the content type of the file being printed matches the content type of the printer. For example, if no -T option is added to the lp

command, **lpsched** assumes that the content type of the file is `simple`. If the printer can accept the `simple` content type, no slow filters are run and **lpsched** passes no filter variable to the model file. This table is used when the content type of the file being printed is not a content type that the printer can accept.

Typical printing scenario

This section presents an example that explains how **lpsched** uses the filter table.

This example uses printer **my_ps**, a serial printer that has a printer type of **PS** and an input content type of **PS**. You issue the following command to print the local **passwd** file:

```
lp -dmy_ps /etc/passwd
```

Without the **-T** option, **lpsched** assumes a document of type `simple`. Since the printer can accept only input that is in type **PS**, the document must be filtered and changed from a `simple` document type to a PostScript file of type **PS**.

The **lpsched** program selects the following filters to process the file:

- L** This filter runs on any printer type, can run on any printer name, has an input type of `simple`, and has an output type of PostScript filter (there is no output type of **PS** from a filter that can accept `simple` as an input type). This is the only entry that can take a `simple` document and convert it to a `postscript` one. The entry runs the **postprint** filter. The output type that this filter produces is type `postscript`. When this filter finishes, the output is not **PS**.
- G** This filter runs on printer type of **PS** (and others), run on any printer name, accept the type `postscript` as input to the filter, and outputs `postdown`. The entry runs the filter called `download` and checks the type Post Script file to see if there are any fonts that need to be downloaded to the printer. Next, **lpsched** places this filtered document in the queue. There is no low filter that can filter an input type of `postdown`.
- A** This is a fast filter, named `postio`, that has an output type of **PS**. No further filtering is necessary. This filter can run on any printer name, accepts `postdown` as input, and releases output in **PS**. This filter is used to communicate with serial PostScript printers interactively. **lpsched** passes the pathname for this filter to the model file by setting the environment variable named `FILTER` to `postio`'s pathname.

The model file then starts the filter `postio` with the filtered file sitting in the print queue as input and sends the filtered file out to the printer. The filtered file in the queue is content type **PS**. The printer requires **PS**, so the file prints correctly.

The system manager can define both slow and fast filters through **sysadm**. These filters are then added to the end of this table and are available for use by **lpsched**.

Remote printing with remshlp

These procedures apply only if you are printing remotely using USL's **lpsched**. They do not apply if you are using BSD's LPD program.

When files do not print on remote machines, first make sure that **lp** works on the local machine (the machine the printer is attached to). Then check the remote connections. Before following the procedures, however, you need to know how **lpsched** arranges printing on a remote machine.

How lpsched prints on a remote machine

When you define a remote printer, you specify a **remshlp** model file. The sample model file below shows how remote printing occurs.

```
result='(
    first_time=y
    for file in ${files}
    do
        if ["$first_time" = "y"]
        then
            first_time=n
        else
            echo "\f\c"
        fi
        /user/bin/cat $file 2>&1
    done
) | /usr/bin/remsh ${host} LOGNAME=${usr} lp -d${printer}
-t\${title} \" -n${copies} ${options} 2>&1'
```

As the example shows, **lpsched** executes a **remsh** command to the remote system with several switches set and the file piped to the **remsh** command from the **cat** command. Because **lpsched** is owned by **lp**, the **remsh** command is logged in as **lp**. This creates some opportunities for troubleshooting.

When troubleshooting printing problems, you can execute manually the commands that the interface script **remshlp_a** runs. This can

help you isolate where the problem occurs. Use the following procedures:

1. Login as root. As a regular user, you cannot login in to the local machine as **lp**, but you can login as root.
2. Issue an **su** command with **lp** as an argument. Root ignores **/etc/passwd** when executing an **su** command.
3. Issue a **remsh** command to the remote machine. For example, **remsh remote host**.

If you are unable to login to the remote host, examine the **/etc/host.equiv** file for the name of the remote host machine. If the remote machine is listed, then the problem must be related to TCP/IP.

If you are able to login to the remote host, try to print the remote host's **/etc/passwd** file with the following command:

```
remsh host lp -dprinter /etc/passwd
```

If the file does not print, try to print the local machine's **/etc/passwd** file. When you are able to print on the local machine, try to print the same file on the remote machine, using the **remsh** command. Use the following command:

```
cat /etc/passwd | remsh host lp -dprinter
```

IMPORTANT: You can use the command as a regular user if you have the same username and password pair on both machines, or if you have a **.rhosts** file in your initial directory with the correct entries.

When the above **cat** command is successful, the remote connection is working properly.

Printing with lpd and lpr

IMPORTANT: We recommend using USL's scheduler **lpsched** (discussed in Chapter 4) because **lpNet** supports remote printing to systems that support either **lpsched** or the **lpd** daemon.

This section is a brief primer on printing with the native BSD print scheduler, which is referred to by its daemon (**lpd**) and print command (**lpr**). The DG/UX system, by default, uses USL's System 5.4 method of printing. You can, however, choose to print using **lpd** and **lpr**. This section lists and defines the programs required and explains how to set up your system to use **lpd** and **lpr**.

Programs and files required

The native BSD method of printing requires the following programs:

lpc	Controls the queues and printers.
lpd	Is the printer daemon.
lpr	Sends files to the printer.
lprm	Removes a job from the queue.
lpq	Examines a queue.
/etc/printcap	Contains description of printers.
/usr/spool/*	Contains spool directories
/usr/spool*/minfree	Specifies the minimum number of free blocks that is to be left by lpd , so that the file system does not fill up. If this file is omitted, the print spool will work, but it could fill up the file system.
/dev/*	Represents the printer device.
/dev/printer	Represents the socket handling local requests.
/etc/hosts.equiv	Lists the machines allowed printer access.
/etc/hosts.lpd	Lists the machines allowed printer access that are not under the same administrative control as the local host.

Setting up BSD printing

When using the BSD method of printing, all local and remote printers must be defined in the **printcap** file. The **printcap** file is similar to the **termcap** file. The **printcap** file lists the printers, the device they are attached to, and any options that are necessary. For details about this file, see the manual page **printcap(5)**.

In the **printcap** file, the first object is a list of names for the printer, separated by the vertical bar (|) character. Next, are various configuration settings separated by colons. Each has the form **XX=string, XX, or XX#number**, where **XX** is the two character name of the option. The *string* and *number* are values assigned to option. You can use filters, but they are generally not necessary for most modern printers. An example of a simple local printer entry in the **printcap** file follows:

```
lp|tlp|ptr:br#1200:fs#06320:sd=/usr/spool/lpd:lp=/dev/tty18:
```

The table below explains the elements in this above entry.

lp tlp ptr:	Names the printer <i>lp</i> with the aliases tlp and ptr .
br#1200	Defines the baud rate of 1200.
fs#06320	Defines the flag set bits of the line. The number 06320 are translated as follows. The 06000 bits indicate that tabs are expanded to spaces. The 00300 bits indicate to accept both even and odd parity on input, which means ignore parity. The 00020 bits indicate to output a linefeed as a carriage–return linefeed pair. See the manual page tty(1) or the header file /usr/include/sys/_int_bsd_tty_ioctl.h . This file defines the values discussed in the manual pages.
Sd=/usr/spool/lpd	Defines the spool directory used by ldp . This file does not have to be located in the /usr/spool directory and can have any valid name. The file, however, must exist. If it does not exist, create it.
Lp=/dev/tty18	Defines the port that the printer is attached to.

The list of objects in the example is not exhaustive. Refer to the manual pages **printcap(5)** and **tty(1)** for a list of all the objects.

Lpc is the administrative program that starts, stops, takes down, and provides the statics of queues. **Lpc** includes the following options:

help <i>command</i> or ? <i>command</i>	Shows you a short list of all commands.
enable <i>printer</i> or disable <i>printer</i>	Enables or disables spooling of jobs to the named printer. Jobs that are already in the queue are not effected.
start <i>printer</i> or stop <i>printer</i>	Enables or disables printing on the named printer. Jobs can still be spooled.
abort	Like stop, but does not allow the active job to complete.
down <i>printer</i> or up <i>printer</i>	This command stops both spooling and printing.
clean	Removes all jobs from the printer's queue including the active job. The active job completes because the printing daemon still has a reference to the file.

topq printer jobid or topq printer <i>username</i>	Moves either the <i>jobid</i> or the files belonging to the user to the top of the queue.
restart printer	Restarts a printing daemon that has mysteriously died.
status printer	Shows whether or not spooling and the printer are enabled. Also, shows the number of entries in the queue and the status of the daemon for that specified printer.

You can execute the above **lpc** commands on the command line. You can also execute **lpc** and specify the commands interactively. Note that the **lpc** program can sometime give unexpected results.

To bring up the **lpd** program, you must set to true the **lpd_START** parameter in the file **/etc/dgux.params**. You must edit the **/etc/dgux.params** file and set the parameter as follows:

lpd_START="true"

You can run either **lpsched** (USL) or **lpd** (BSD) or both. The command **/usr/sbin/rc.sched** brings up the scheduler(s) you select. Do not, however, assign a the same printer to both schedulers. This could lead to some confusion between the schedulers.

When you have edited the **/etc/printcap** file and executed the command **lpc up all**, you should be able to print using BSD.

Example

This section presents an example of how to set up a BSD printer. Remember BSD printers are not related to **lpsched** or printer management in **sysadm**.

This example sets up a printer by the name of **lpt** with the following attributes:

- Resides on tty line 16.
- Uses odd parity.
- Expects 8 bit data.
- Operates on 9600 baud.
- Uses XON/XOFF software flow control.

The entry in the **printcap** file would be as follows:

```
lpt | ptr:br#9600:
```

The first field is the printer name (**lpt**) and alias (**ptr**). The fields are separated by a colon (:). The next field is the baud rate (**br**), set to 9600.

When setting up a BSD printer, you must turn on the characteristics you want and turn off the ones you do not. The **fs** field is the set flags field and the **fc** field is the clear field flag.

End of Appendix

B Determining tty lines for asynchronous ports

This appendix explains how to gather information for adding printers and terminals to the DG/UX system. It provides both sample and blank worksheets to aid in recording information.

In this appendix, the expression *terminal line controller* refers to the RS-232/422 interfaces for the ports on the computer unit, the VAC/16 controller, and the VDA/128 and VDA/255 host adapters. The term “ports” on a VDA host adapter refers to the ports on the cluster controllers that are connected to the host adapter.

Major sections in this appendix are:

- Port tty line numbers
- Completing terminal controller worksheets
- Determining the tty line for terminal line controller ports

Port tty line numbers

The DG/UX system automatically assigns a tty line number to each attached port in the hardware configuration when the system boots. A tty line number takes the form:

`tt x`

where x is a sequentially assigned number. For example, **tt 00** refers to the first port, **tt 01** the second port, and so on. A file with the name of the tty line number is created in the `/dev` directory each time the system boots.

If you have ports attached to multiple, different terminal line controllers, you must determine each port’s tty line number — as explained next.

You can add printers and terminals all at once or one at a time. When you add all terminals or printers at once, you assign the same characteristics to all asynchronous ports on your computer. This includes RS–233/422 ports on the computer, ports on any Systech asynchronous controllers, and ports on cluster controllers for any Systech distributed host adapters.

When adding printers or terminals one at a time, you can assign different characteristics to different ports. See Chapters 3 through 6 for information on adding printers through **sysadm**.

To add a terminal to your DG/UX system, you must know

- The tty line number that the DG/UX system assigned to the terminal line controller port where the printer is connected. This number depends on the terminal controller, number and type of previous terminal controllers, and port number on the controller. To learn it, you may need to use the AViiON system diagnostics to perform the following tasks:
 - a. Find the order of controller names in your system file.
 - b. Learn the terminal line controller type and cluster controller type
 - c. Learn the port where each device is connected

After learning the tty line number(s) for each terminal controller, you will complete a worksheet so you won't need to perform these tasks again later.

- The type of printer connected to the port (for example, PostScript). You will then complete another worksheet for all these devices.

After gathering this information and completing the worksheets, you can then add printers using **sysadm**.

Completing terminal controller worksheets

This section contains samples and blank originals of the following worksheets:

- Terminal Line Controllers Worksheet
- RS/232/422 Ports on the Computer Unit Worksheet
- VAC/16 Controller Worksheet
- VDA Host Adapter Worksheet

Each VDA Host Adapter Worksheet contains space for recording 32 ports. So, if you have a VDA host adapter, you will need two additional sheets for a VDA/128 host adapter and six additional sheets for a VDA/255 host adapter.

Preceding each worksheet is a sample worksheet that has been filled out for an AViiON 5000 computer with three terminal line controllers: RS-232/422 terminal/modem port on the computer unit, one VAC/16 controller, and one VDA/128 host adapter. The host adapter has one 8-line cluster controller and seven 16-line cluster controllers, providing ports for a maximum of 120 serial devices. These terminal line controllers have the device names given below, and these device names are listed in your system file in the relative order shown below:

duart()	terminal/modem port on computer unit
syac()	VAC/16
syac(1)	VDA/128

Figures B-1 and B-2 show a sample terminal line controllers worksheet and a copy for you to complete.

Figures B-3 and B-4 show a sample RS-232/422 ports on computer unit worksheet and a copy for you to complete.

Figures B-5 and B-6 show a sample VAC/16 controller worksheet and a copy for you to complete.

Figures B-7 and B-8 show a sample VDA host adapter worksheet and a copy for you to complete.

Appendix C contains additional copies of the blank worksheets.

Sample Worksheet

Terminal Line Controllers Worksheet

Board No.	Device Name	Configuration File Position	Board or Port Type	Cluster Controllers			
				Address	No. Lines	Address	No. Lines
	duart()	1					
	duart(1)						
0	syac()	2	VAC / 16	01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
				01	8	09	
1	syac(1)	3	VDA / 128	02	16	0A	
				03	16	0B	
				04	16	0C	
				05	16	0D	
				06	16	0E	
				07	16	0F	
				08	16	10	
				01		09	
				2	syac(2)		
03		0B					
04		0C					
05		0D					
06		0E					
07		0F					
08		10					
01		09					
3	syac(3)						
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
				01		09	
				4	syac(4)		
03		0B					
04		0C					
05		0D					
06		0E					
07		0F					
08		10					
01		09					

Figure B-1 Sample worksheet for terminal line controllers

Terminal Line Controllers Worksheet

Board No.	Device Name	Configuration File Position	Board or Port Type	Cluster Controllers			
				Address	No. Lines	Address	No. Lines
	duart()	1					
	duart(1)						
0	syac()	2		01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
1	syac(1)	3		01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
2	syac(2)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
3	syac(3)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
4	syac(4)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	

Figure B-2 Worksheet for terminal line controllers

Sample Worksheet

Computer Unit RS-232/422 Port Worksheet

Port type: <i>terminal / modem</i>		Device name: <i>duart()</i>
tty Line	Device Type	Description
<i>oo</i>	<i>modem to</i>	<i>lab B2, conn 2203</i>
Port type:		Device name: <i>duart(1)</i>
tty Line	Device Type	Description

Figure B-3 Sample Worksheet for RS-232/422 Ports On Computer Unit

Computer Unit RS-232/422 Port Worksheet

Port type:		Device name: <i>duart()</i>
tty Line	Device Type	Description
Port type:		Device name: <i>duart(1)</i>
tty Line	Device Type	Description

Figure B-4 Worksheet for RS-232/422 Ports On Computer Unit

Sample Worksheet

VAC/16 Controller Worksheet

Board no: 0				Device name: <i>syac()</i>				Range of tty lines: 01 -16			
Port No.	tty Line	Device Type	Description	Port No.	tty Line	Device Type	Description				
0	01	4558 printer	officeA1 conn 1100	8	09	D216+	officeA10 conn 1118				
1	02	Epson printer	officeA2 conn 1102	9	10	VT100	officeA11 conn 1120				
2	03	D216+	officeA3 conn 1104	10	11	D462+	officeA12 conn 1122				
3	04	D462+	officeA4 conn 1106	11	12	D462+	officeA14 conn 1124				
4	05	D216+	officeA5 conn 1108	12	13	D462+	officeA14 conn 1124				
5	06	D216+	officeA6 conn 1110	13	14	D462+	officeA18 conn 1128				
6	07	D216+	officeA5 conn 1114	14	15	D462+	officeA20 conn 1130				
7	08	D216+	officeA5 conn 1116	15	16	D462+	officeA21 conn 1132				

Figure B-5 Sample Worksheet for a VAC/16 Controller

VAC/16 Controller Worksheet

Board no: 0				Device name: <i>syac()</i>				Range of tty lines: 01 -16			
Port No.	tty Line	Device Type	Description	Port No.	tty Line	Device Type	Description				
0				8							
1				9							
2				10							
3				11							
4				12							
5				13							
6				14							
7				15							

Figure B-6 Worksheet for a VAC/16 Controller

Sample Worksheet

VDA Host Adapter Worksheet

Sheet 1 of 4

Board type: <i>VDA/128</i> Board no: <i>1</i> Device name: <i>syac(1)</i> Range of tty lines: <i>17-271</i>									
Cluster Address	Port No.	tty Line	Device Type	Description	Cluster Address	Port No.	tty Line	Device Type	Description
01	0	17	D216+	office B1,conn 1200	02	0	33	VT100	office B9,conn 1216
	1	18	D216+	office B2,conn 1202		1	34	VT100	office B10,conn 1218
	2	19	D462+	office B3,conn 1204		2	35	D216+	office B11,conn 1220
	3	20	VT100	office B4,conn 1206		3	36	D462+	office B10,conn 1222
	4	21	VT100	office B5,conn 1208		4	37	D462+	office B13,conn 1224
	5	22	VT100	office B6,conn 1210		5	38	VT100	office B14,conn 1226
	6	23	D216+	office B7,conn 1212		6	39	D462+	office B15,conn 1228
	7	24				7	40	D462+	office B16,conn 1230
	8	25	6772ptr	lab B2,conn 2204	8	41	D462+	office B17,conn 1232	
	9				9	42	D462+	office B18,conn 1234	
	10				10	43	D462+	office B19,conn 1236	
	11				11	44	D462+	office B20,conn 1238	
	12				12	45	D462+	office B21,conn 1240	
	13				13	46	D462+	office B22,conn 1242	
	14				14	47	D462+		
	15				15	48	D462+		

Figure B-7 Sample Worksheet for a VDA Host Adapter

VDA Host Adapter Worksheet
Sheet ____ of ____

Board type:				Board no:	Device name	Range of tty lines:			
Cluster Address	Port No.	tty Line	Device Type	Description	Cluster Address	Port No.	tty Line	Device Type	Description
	0					0			
	1					1			
	2					2			
	3					3			
	4					4			
	5					5			
	6					6			
	7					7			
	8					8			
	9					9			
	10					10			
	11					11			
	12					12			
	13					13			
	14					14			
	15					15			

Figure B-8 Worksheet for a VDA Host Adapter

Determining the order of DG/UX device names for terminal line controllers

You will need to know the relative order of the terminal line controller device names in your system file.

Change to the current directory of the system file and use the **more** command to view its contents. An example follows:

```
# cd /usr/src/uts/aviion/Build ↵
# more system.aviion ↵
```

IMPORTANT: You may have assigned a different name to your system file. View the appropriate one. Refer to *Managing the DG/UX™ System* for information on building kernels and editing system files.

Figure B-9 shows an excerpt from a system file named **system.aviion**

```
#-----
# Automatically Configured Hardware Devices:
#
# These hardware devices were found on the system by probedev(1M).
#
#   kbd()           ## Workstation keyboard
#   grfx()          ## Workstation graphics display
#   lp()            ## Integrated parallel line printer controller
#   duart(0)        ## Dual-line terminal controller (number 0)
#   inen()          ## Integrated Ethernet controller
#   syac()          ## Systech terminal line controller
#   syac(1)         ## second Systech terminal line controller
#   sd(incsc(),0)   ## SCSI disk 0 on Integrated SCSI adapter
#   sd(incsc(),1)   ## SCSI disk 1 on Integrated SCSI adapter
#   st(incsc(),4)   ## SCSI tape 4 on Integrated SCSI adapter
#   st(incsc(),6)   ## SCSI tape 6 on Integrated SCSI adapter
```

Figure B-9 Automatically configured devices in the system configuration file

Viewing the system file, you can see the relative order of the terminal line controllers. The ordering follows:

duart(0)	position 1
syac()	position 2
syac(1)	position 3

The order in which terminal line controllers are configured and listed in the system file is important because it affects the

assignment of tty line numbers. Be sure to record the relative order of your terminal line controller devices.

Learning the terminal line controller type and cluster controller type

This section describes how to use your AViiON System Diagnostics to help you get the following information:

- Board type (VAC/16, VDA/128, or VDA/255) for the **syac** terminal line controller device name.
- Cluster controller type (8-line, 16-line) for each controller connected to a VDA host adapter.
- VAC/16 or cluster controller port to which a specific terminal or printer is connected.

IMPORTANT: The person installing your computer hardware should have recorded this information on Device Worksheets that were supplied in the manual *Setting Up and Installing VMEbus Options in AViiON® Systems*. If these worksheets are available from the hardware installer and they have all the information listed above, then go to the section “Determining the tty Line for Terminal Line Controller Ports” later in this appendix. If these worksheets are not available or they are missing information, continue with this section.

If your AViiON computer was preloaded with the DG/UX system software, the system diagnostics is also on the boot disk. You boot the diagnostics as a stand-alone program. If your computer came with the DG/UX system on a release tape, you can boot the diagnostics from the AViiON System Diagnostics cartridge tape.

To boot the system diagnostics, you must first shut down the DG/UX system. If the SCM prompt appears on the screen, the DG/UX system is shut down and you should go to the appropriate section below for booting the diagnostics from disk or tape. If the DG/UX system is running, shut it down as described below.

Shutting down your DG/UX system

Before you perform a shutdown, notify all system users of your intention and that they must log out. You may send users a broadcast message to warn of the shutdown and give a grace period. Make sure all users are logged out before proceeding.

From the system console at the shell prompt, type these commands:

```
# cd / ↵
# shutdown -g30 -y ↵
```

The first command changes the current directory to / (root). The second command specifies a grace period of 30 seconds between sending an automatic message to users logged in warning them of the shutdown and the beginning of the shutdown. Also, it includes an affirmative response (yes) to start the shutdown. Otherwise, a confirmation request would be explicitly issued, requiring a response. You will see the following messages:

```
Shutdown started.           Wed June 16 11:08:57 EST
1991
.
.
.
Shutdown is complete.
```

Your system has been shut down.

You next type:

```
# halt -q ↵
```

This command halts the processors, by-passing a confirmation request. After the SCM prompt is displayed, you will know that the DG/UX system has shut down.

Go to the appropriate section that follows for booting the diagnostics from disk or tape.

Booting system diagnostics from disk

To boot the system diagnostics from disk, use the following command syntax:

```
b disk-nameusr:/stand/diags
```

where:

- b** stands for boot.
- disk-name** is the DG/UX device name for the boot disk.
- usr:/stand/diag** specifies virtual disk **usr** which contains the bootable diagnostics program that is located in the file named **/stand/diags**.

CAUTION: *If the **usr** virtual disk were created in pieces, you cannot boot system diagnostics from disk. You must, instead, boot from the release tape; those instructions are given in the next section.*

A typical command for booting system diagnostics from disk follows:

```
SCM> b sd(insic(),0)usr:/stand/diags ↵
```

Table B-1 Default local boot device and system disk

AViiON Series	Default Boot Device		Default System Disk
	CD-ROM	Tape	
100, 200, 300, 400, 3000, 4000, 4300	sd(insic(0),3)	st(insic(0),4)	sd(insic(0),0,0)
5000, 6000, 7000, 8000	sd(cisc(0),3)	st(cisc(0),4)	sd(cisc(0),0,0)
5000, 6000, 7000, 8000	sd(dgsc(0),3)	st(dgsc(0),4)	sd(dgsc(0),0,0)
7000, 8000	sd(hada(0),3)	st(hada(0),4)	sd(hada(0),0,0)
550, 4600, 7400, 8400, 8500, 9500, 5500, 550, 500	sd(ncsc(0),3)	st(ncsc(0),4)	sd(ncsc(0),0,0)

If you do not know the boot device names for your system, consult your system’s hardware installation manual or the equipment packing list . *Managing Mass Storage Devices and DG/UX™ File Systems* provides detailed information on naming devices.

Go to the section “Determining controller types” below.

Booting system diagnostics from tape

To boot the system diagnostics from tape, use the following command syntax:

b *tape-name*

where:

b stands for boot.

tape-name is the DG/UX device name for the boot disk. Table B-1 contains a list of default boot tape (and disk) devices. Select one that is appropriate.

A typical command for booting system diagnostics from tape follows:

```
SCM> b st(insc(),4) ↵
```

Go to the section “Determining controller types” below.

Determining controller types

Figure B–10 shows the first screen of the System Diagnostics:

```
Initializing Operating System
Sizing Optional Controllers:
VME SCSI Controller at 0xffffc000          Found
VME SCSI Controller at 0xffffc080          Found
VME LAN Controller at 0xffff4000           Found
VME Synchronous Controller at 0x55b00000    Found
VME Asynchronous Controller at 0x60000000   Found
VME Asynchronous Controller at 0x60020000   Found
```

Figure B–10 First screen of system diagnostics

Information about the licensing of the diagnostics appears first. When you are ready to continue, proceed as follows:

1. Press Enter to clear the screen. The system diagnostics program looks for (sizes) components in your computer, and displays information about each component that it finds, as shown in the upper portion of the sample screen that follows.
2. Next, the diagnostics initializes the components that it found and displays the prompts in the lower portion of the sample screen below. The display on your screen may differ.

```
16384 Kbytes system memory
15109 Kbytes memory available for test
PROM revision xx.xx
Dual CPU System (Motorola 881x0 CPU Rev x)
2 Instruction Caches (Motorola 88xx0 CMMU Rev x)
2 Data Caches (Motorola 8xx00 CMMU Rev x)
Initializing Virtual Console
Initializing Real Time Clock
Initializing VME Async I/O Board 0
Initializing VME Async I/O Board 1
Initializing VME SCSI Controller 0
Initializing VME SCSI Controller 1
Initializing VME SCSI Controller 2
Initializing VME SCSI Controller 3
Initializing Duart
Initializing VME LAN Board 0
Initializing VME LAN Board 1
Initializing VME Sync I/O Board 0

Current time is 16:24 Monday, January 25, 1993.
Is this correct (y/n) [y]?
```

3. Verify that the time displayed is correct. Next, the system diagnostics lists the peripheral devices connected to initialized controllers, as shown in the sample screen that follows.

Current time is xx:xx Tuesday, May 4, 1993. Is this correct (Y/N)
[Y]?

Sizing Peripherals....

VME SCSI Board 0:

Unit 0: Microp 1578-15 UPDG02 Disk Drive found
Unit 1: Microp 1578-15 UPDG02 Disk Drive found
Unit 3: TEAC 5.25 Floppy (LUN 0) Disk Drive found
Unit 3: TEAC 5.25 Floppy (LUN 1) Disk Drive found
Unit 4: Archive Viper 150 21247-045 Tape Drive found

VME SCSI Board 1:

Unit (Drive Number) 0: 662 MB ESDI Disk found
Unit (Drive Number) 1: 662 MB ESDI Disk found
Unit (Drive Number) 2: 662 MB ESDI Disk found

VME Async Board 0:

128-line VME Host Adapter
Model = HPS-6945
Firmware P/N = 90-070052-3-02A

Sizing Cluster Controller Network

Net ID = 01 (hex): HPS-7088-020 (Ready)
Net ID = 08 (hex): HPS-7088-020 (Ready)
Net ID = 13 (hex): HPS-7082-020 (Ready)

VME Async Board 1:

16-line VME Async Board
Model = HPS-6236
Firmware P/N = 90-070408-8-01A

Press New Line to proceed

Record each VME async board type listed under “Sizing Peripherals....” The names that the diagnostics system uses for the different terminal line controller boards are as follows:

16-line Async Board	VAC/16
128-line VME Host Adapter	VDA/128
255-Line VME Host Adapter	VDA/255

For example, using the sample screen above, you would write "VDA/16" in the Board or Port Type column for Board No. 0, and "VAC/128" in the Board or Port Type column for Board No. 1.

Record the number of lines for each cluster controller address (Net ID) listed under a VME host adapter. The model numbers for the 8-line and 16-line cluster controllers are as follows:

```
HPS-7082      8-line cluster controller
HPS-7088      16-line cluster controller
```

For example, using the sample screen above, for Board 0, under "Cluster Controllers" you would write "8" in the "No. Lines" column for Address 01, and write "16" in the "No. Lines" column for Addresses 02 and 08.

When you have finished recording this information, press Enter. If more sizing information appears, repeat these procedures appropriately. Continue this process of viewing and recording sizing information until the Main Menu appears.

Before continuing, you should transfer information from the Terminal Line Controllers Worksheet to the tty worksheets as described below. These worksheets are located in Appendix C. If you do not do this, you will have difficulty determining the tty line assigned to each port on these terminal line controllers.

VAC/16

For each VAC/16 controller in your computer, record the board number and its device name on a Device Worksheet for a VAC/16 Controller.

VDA Host Adapter

For each VDA host adapter in your computer, record its board number, device name, and device type (VDA/128 or VDA/255) on a Device Worksheet for a VDA Host Adapter. For each cluster controller connected to the host adapter, record the cluster address. If the cluster controller is an 8-line controller, draw a vertical arrow from the cluster address you entered down to the dashed line. This indicates that only nine ports are available on this controller. If the controller is a 16-line controller, draw the vertical line through the dashed line all the way down to the bottom of the column to indicate that 16 ports are available.

You now have enough information to determine which tty line your DG/UX system assigns to a specific port on any of the terminal line controllers in your computer. In addition, if you know what type of terminal or printer is connected to each terminal line controller

port, you can add these devices to your DG/UX system without having to complete the following section. In this case, you should record the device type for each port on the appropriate tty worksheet.

Next, exit to the SCM by selecting option 4 on the Main Menu, and go to the “Determining the tty Line for Each Terminal Line Controller Port” section later in this appendix. If you do not know which type of terminal or printer is connected to each terminal line controller port, continue on to the next section.

Determining the port where a specific device is connected

While you do not need to know the port to which a specific device is connected to determine the tty line for that port, you will need this information when you add a terminal or printer to your DG/UX system, or if you need to troubleshoot problems with a terminal line controller. This section will help you obtain that information.

The following figure shows the System Diagnostics’ Main Menu that appears on the system console.

```

System Diagnostics
Revision: xx.xx

Data General Corporation
Proprietary Use Only

Main Menu

1. Run Acceptance test
2. View Tools Menu
3. Display help screen
4. Exit to SCM

Enter choice [1]:
    
```

1. At the Main Menu, select **2** from the View Tools Menu and press Enter.

The Tools Menu shown below appears.

```

Tools Menu
1. Format diskettes
2. Run tape adjustment utility
3. View Graphics Tools Menu
4. Test network connection (TDR)
5. Run keyboard test
6. Run mouse test
7. View Terminal Test Menu
8. Display help screen
9. Return to main menu

Enter choice [9]:
    
```

2. Select 7 from the View Terminal Test menu and press Enter.

The Terminal Test Menu shown below appears

```

Terminal Test Menu
1. Start scrolling character set test
2. Start lines of characters test
3. Start keyboard echo test
4. Start port ID message test
5. Auto port identification
6. Terminate a test
7. Show executing tests
8. Display help screen
9. Return to Tools menu

Enter choice [9]:
    
```

**3. Select 4 from the Start port ID message test and then press Enter.
The system displays the following prompt:**

```

Board number (0,1, [ALL])?

Running selftest on VME Host Adapter 0 (approximately 30 seconds),
please wait....
    
```

4. Press Enter

This message occurs the first time you select a host adapter for testing. On each terminal or printer connected to a port on a

VAC/16 controller or VDA host adapter in your computer, you will see a port ID message similar to the one below. This message lists the board number, cluster address (VDA host adapter only), and the port number for the device displaying the message.

```
128-line VME Host Adapter 0, Cluster address: 01,  
port: 0
```

IMPORTANT: If you have an 8-line cluster controller without a parallel printer connected to port 8, or the printer is not on line and ready, a message appears telling you this. If such a message appears, simply press the Esc key to skip the test on that port.

5. Look at the message displayed on each device for which you do not know the port number, and determine the board number, cluster address, and port number for the device. On the appropriate Device Worksheet under the specified board number, cluster address (if applicable), and port number, record the type of device (for example, D460 terminal or Model 6640 parallel laser printer) displaying the message. Also record a description that locates the device (for example, office 3B, connector #1356). While the port ID messages are being displayed, the Terminal Test Menu appears on the system console screen.
6. After recording the information for each terminal or printer, select **6** from the Terminal test menu and then press Enter. The following prompt appears:

```
Board number (0,1, [ALL])?
```

7. Press Enter to accept the default response to the next prompt: From the Terminal Test Menu, press Enter again to select the Return to Tools menu.
8. From the Tools Menu, press Enter to select Return to main menu.
9. At the Main Menu, select **4** to exit to the SCM followed by Enter.

Determining the tty line for terminal line controller ports

This section discusses how the DG/UX system allocates tty lines to asynchronous terminal line controller ports. With this information, you can determine the tty lines that your DG/UX system assigns to each such port on your computer.

How the DG/UX system allocates tty lines

When you booted your machine, it automatically allocated a specific tty line for each port on each terminal line controller in your

computer. Table B-2 lists the number of tty lines that the DG/UX system allocates to each type of terminal line controller.

Table B-2 Number of tty Lines Allocated to Terminal Line Controllers

Terminal Line Controller	Lines Allocated
RS-232/422 ports on computer unit	1
VAC/16 controller	16
VDA/128, VDA/255, or VTC/256 host adapter	256

Notice that the DG/UX system allocates 256 tty lines to a VDA/128 host adapter. Since a VDA/128 host adapter has only 128 ports, this means that only the first 128 tty lines are actually assigned to specific ports on a VDA/128; the remaining 128 tty lines are unused.

The DG/UX system assigns a specific tty line to each port sequentially in the order in which the names of the terminal line controllers are listed in your system file. It starts with **tty00**. For each subsequent port, the numerical portion of the tty name is increased by one.

Let's look at an example of how the DG/UX system assigns tty lines. The system file lists the following terminal line controllers in the order shown below:

```
duart()    ## integrated Duart terminal line controller
syac()    ## first Systech terminal line controller
syac(1)   ## second Systech terminal line controller
```

Further suppose that **syac()** is a VAC/16 controller and **syac(1)** is a VDA/128 host adapter. For this configuration, the DG/UX system assigns tty lines as follows:

```
tty00 to the RS-232/422 port on the duart.
tty01–tty16 to the VAC/16 controller.
tty17–tty272 to the VDA/128 host adapter.
```

If the **duart** was listed after the two **syac** devices, the tty line assignment would be as follows:

```
tty00–tty15 to the VAC/16 controller.
tty16–tty271 to the VDA/128 host adapter.
tty272 to the RS-232/422 port on the duart.
```

The DG/UX system assigns each of the 16 tty lines that it allocates to a VAC/16 host adapter to its 16 ports in sequential order. In other

words, in the example above where **tty01** through **tty16** are allocated to the VAC/16 controller, **tty01** is assigned to port 0, **tty02** is assigned to port 1, **tty03** is assigned to port 2, and so on.

Since devices connect to a VDA host adapter through ports on cluster controllers, the DG/UX system allocates specific tty lines to those ports. Table B-3 shows how the DG/UX system allocates 16 tty lines to each cluster controller address, 01 through 10 hexadecimal (16 decimal). This means that the DG/UX system allocates 16 tty lines to an 8-line cluster controller with one of these addresses. Since an 8-line cluster controller has eight asynchronous ports (ports 0 through 7) and one parallel printer port (port 8), the last seven tty lines allocated to this controller's address are unused. The DG/UX system assigns the 16 tty lines that it allocates to a cluster controller address as follows: the tty line with the lowest number (call it n) to port 0; the next tty line with the next highest number ($n+1$) to port 1; the one with the next highest number ($n+2$) to port 2, and so on.

Table B-3 tty Lines Allocated to Cluster Controller Addresses

Cluster Controller Address	tty Lines Allocated
01	tty (n) through tty ($n+15$)
02	tty ($n + 16$) through tty ($n + 31$)
03	tty ($n + 32$) through tty ($n + 47$)
04	tty ($n + 48$) through tty ($n + 64$)
05	tty ($n + 64$) through tty ($n + 79$)
06	tty ($n + 80$) through tty ($n + 95$)
07	tty ($n + 96$) through tty ($n + 111$)
08	tty ($n + 112$) through tty ($n + 127$)
09	tty ($n + 128$) through tty ($n + 143$)
0A	tty ($n + 144$) through tty ($n + 159$)
0B	tty ($n + 160$) through tty ($n + 175$)
0C	tty ($n + 176$) through tty ($n + 191$)
0D	tty ($n + 192$) through tty ($n + 207$)
0E	tty ($n + 208$) through tty ($n + 223$)
0F	tty ($n + 224$) through tty ($n + 239$)
10	tty ($n + 240$) through tty ($n + 255$)

IMPORTANT: The cluster controller address is also called the node address. For more information on these addresses, refer to the *HPS Downloadable Cluster Controller Installation Guide*. The last tty line, **tty**($n+255$), allocated to the cluster controller with address 10 is not used. If this cluster controller is a 16-line controller, this tty line is assigned port 15, so cannot be used. If it is an 8-line cluster

box, this tty line is one of the seven unused tty lines allocated to the controller.

Continuing with this example, let's assume that the VDA/128 host adapter with one 8-line cluster controller and one 16-line cluster controller is allocated tty lines **tty16** through **tty270**. If the 8-line cluster controller has address 01 and the 16-line controller has address 02, then the tty lines for the ports on the controllers are as follows.

Examples:

8-line cluster controller (address 01):

tty16 through **tty24** for ports 0 through 8 (port 8 is the parallel printer port).

tty25 through **tty31** are unused.

16-line cluster controller (address 02):

tty32 through **tty47** for ports 0 through 15.

tty48 through **tty271** are unused.

You should now have enough information on your Terminal Line Controller Worksheet and your tty worksheets to determine the specific tty line that your DG/UX system assigned to each port on a terminal line controller. Using these worksheets, proceed as follows:

1. Using the Terminal Line Controller Worksheet that you completed in Appendix C, find the device name with Configuration File Position 1, then get the tty worksheet that has that device name written on it.
2. Determine the tty line or range of tty lines assigned to this device name using the procedure below. Note that the AViiON 4600 series machines have three asynchronous ports, which could be **tty0**, **tty1**, or **tty2**.

If the device name is **duart()** or **duart(1)** write "00" in the tty Line column on the tty worksheet for that device name. If the device name is **syac()**, **syac(1)**, **syac(2)**, **syac(3)**, or **syac(4)**, use the formula in Table B-4 for the board type of that device name to calculate the range of tty lines assigned to that board (where $n = 00$), and record this range on the tty worksheet for the board.

3. Using the Terminal Line Controller Worksheet, find the device name with previous higher Configuration File Position, and get the tty worksheet that has that device name written on it.
4. Determine the tty line or range of tty lines assigned to this device as described below. Use the following value for n :

$$n = 1 + [\textit{highest tty line number you calculated for next device name}]$$

If the device name is **duart()** or **duart(1)**, write the value for *n* in the tty Line column on the tty worksheet for that device name. If the device name is **syac()**, **syac(1)**, **syac(2)**, **syac(3)**, or **syac(4)**, use the formula in the next table for the board type of that device name to calculate the range of tty lines assigned to that board, and record this range on the tty worksheet for the board.

Table B-4 Lines Allocated to Systech Terminal Line Controllers and Cluster Controllers

Board Type	Range of tty Lines Allocated
VAC/16	tty(<i>n</i>) through tty(<i>n</i> + 15)
VDA/128 or VDA/255	tty(<i>n</i>) through tty(<i>n</i> + 255)
Cluster controller with address:	
01	tty(<i>n</i>) through tty(<i>n</i>+15)
02	tty(<i>n</i> + 16) through tty(<i>n</i> + 31)
03	tty(<i>n</i> + 32) through tty(<i>n</i> + 47)
04	tty(<i>n</i> + 48) through tty(<i>n</i> + 64)
05	tty(<i>n</i> + 64) through tty(<i>n</i> + 79)
06	tty(<i>n</i> + 80) through tty(<i>n</i> + 95)
07	tty(<i>n</i> + 96) through tty(<i>n</i> + 111)
08	tty(<i>n</i> + 112) through tty(<i>n</i> + 127)
09	tty(<i>n</i> + 128) through tty(<i>n</i> + 143)
0A	tty(<i>n</i> + 144) through tty(<i>n</i> + 159)
0B	tty(<i>n</i> + 160) through tty(<i>n</i> + 175)
0C	tty(<i>n</i> + 176) through tty(<i>n</i> + 191)
0D	tty(<i>n</i> + 192) through tty(<i>n</i> + 207)
0E	tty(<i>n</i> + 208) through tty(<i>n</i> + 223)
0F	tty(<i>n</i> + 224) through tty(<i>n</i>+239)
10	tty(<i>n</i> + 240) through tty(<i>n</i> + 255)

IMPORTANT: *n* = the lowest tty line number assigned to the board type.

5. Repeat Steps 3 and 4 to determine the tty line or range of tty lines assigned to any other terminal line controllers in your computer. If your computer contains a VAC/16 controller, continue to Step 6; otherwise, go to Step 7.
6. On the tty worksheet for each VAC/16 controller, write the appropriate tty line in the tty Line column for each port number. The lowest numbered tty line allocated to the controller is the one for port 0, the next higher numbered tty line allocated is the one for port 1, and so on.

7. If your computer contains a VDA host adapter, continue to Step 8; if not, then you have finished determining the specific tty line that the DG/UX system assigns to each terminal line controller port in your system.
8. On the tty worksheet for each VDA host adapter, write the appropriate tty line number in the tty Line column for each port number for each cluster address. The lowest numbered tty line allocated to the controller is the one for port 0 on the cluster controller with address 01, and the next higher numbered tty line allocated is the one for port 1 on the same cluster controller, and so on. Since the DG/UX system assigns 16 tty lines to each cluster controller regardless of the number of ports it has, the seven highest tty lines assigned to an 8-line cluster controller are not used. To determine the tty line for port 0 on successive cluster controllers, use the formulas in the preceding table.

You have finished determining the specific tty line that the DG/UX system assigns to each terminal line controller port in your system.

End of Appendix

C Printer Worksheets

This appendix contains the following worksheets for recording information about local printers (attached directly to your AViiON computer) and remote printers (attached to another AViiON computer, but accessible via a LAN):

- Local Printer Planning Worksheet
- Remote Printer Planning Worksheet
- Terminal Line Controllers Worksheet
- RS/232/422 Ports on the Computer Unit Worksheet
- VAC/16 Controller Worksheet
- VDA Host Adapter Worksheet

Terminal Line Controllers Worksheet

Board No.	Device Name	Configuration File Position	Board or Port Type	Cluster Controllers			
				Address	No. Lines	Address	No. Lines
	duart()	1					
	duart(1)						
0	syac()	2		01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
1	syac(1)	3		01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
2	syac(2)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
3	syac(3)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	
4	syac(4)			01		09	
				02		0A	
				03		0B	
				04		0C	
				05		0D	
				06		0E	
				07		0F	
				08		10	

Computer Unit RS-232/422 Port Worksheet

Port type:		Device name: <i>duart()</i>
tty Line	Device Type	Description
Port type:		Device name: <i>duart(1)</i>
tty Line	Device Type	Description

VAC/16 Controller Worksheet

Board no: <i>0</i>		Device name: <i>syac()</i>		Range of tty lines: <i>01-16</i>			
Port No.	tty Line	Device Type	Description	Port No.	tty Line	Device Type	Description
0				8			
1				9			
2				10			
3				11			
4				12			
5				13			
6				14			
7				15			

VDA Host Adapter Worksheet
Sheet ____ of ____

		Board type:			Board no:	Device name	Range of tty lines:			
Cluster Address	Port No.	tty Line	Device Type	Description		Cluster Address	Port No.	tty Line	Device Type	Description
	0						0			
	1						1			
	2						2			
	3						3			
	4						4			
	5						5			
	6						6			
	7						7			
	8						8			
	9						9			
	10						10			
	11						11			
	12						12			
	13						13			
	14						14			
	15						15			

End of Appendix

Index

Symbols

`/dev/lp`, parallel ports, 1-2
`/dev/lp` port
 autopush command, A-28
 parallel printing, 3-2
 passing stty options to streams
 modules, A-28
`/dev/tty`, serial connections, 1-3
`/etc/tcload/vtc.addrs` file, setting up
 printers accessed through the VTC,
 2-12
`/usr` directory, `/usr/lib/kbd/iconv_data`,
 9-58
`/var` directory, `/var/spool/lp/log`, 9-32

A

accept command, 8-33
 accepting print requests, 9-25
 example using the, 9-26
Accepting print requests
 accept command, 8-33
 example of, 8-33
Accessing remote printers
 `/etc/hosts.equiv`, 5-2
 sysadm, 5-2
Accessing SunOS systems, from DG/UX,
 4-14
Adding
 printer classes, 7-23
 printer devices, 7-3, 7-10, 8-4, 8-31
 printer filters, 7-26
 printer forms, 7-30
 remote classes, 7-17
 remote printers, 7-17, 7-38
Adding a printer to `/dev/lp`
 example of, 3-4
 sysadm, 3-4
Adding an HP LaserJet 4 printer
 assigning IP addresses, 6-7
 gathering information, 6-7
 using sysadm, 6-7
Adding filters, example of, 9-22
Adding HP Laserjet 4 Printers, example
 of, 6-9
Adding HP LaserJet printers, sysadm,
 6-9
Adding local printers
 `/dev/tty`, 1-3
 connection method, 8-6
 example of, 8-31
 interface program, 8-10
 lpadmin command, 8-5
 printer name, 8-5
Adding NetWare machines, on local
 DG/UX systems, 4-8
Adding NetWare machines on DG/UX
 systems, sysadm, 4-8
Adding printers
 attached to Novell NetWare systems,
 4-6
 connected to DG/UX 5.4 hosts, 4-3,
 7-17
 connected to FastPort print servers,
 2-14
 connected to SunOS systems, 4-13,
 4-16
 datability termservers, 6-5
 example of, 4-3
 from a BSD system, 5-3
 from a remote pre-5.4 host, 5-3
 from BSD systems, 5-1, 5-14
 from other vendors, 4-1
 from pre-5.4 systems, 5-1, 5-14
 from remote AOS/VS systems, 5-9
 from SunOS systems, 4-14
 full add feature, 3-1, 7-4
 lpNet, 4-4
 quick add feature, 3-1, 7-4
 remote, 4-12
 remote from AOS/VS systems, 5-7
 remsh, 5-4, 5-15
 sysadm, 4-4, 4-19
 to termservers, 6-2, 6-4
 using the lpadmin command, 8-31
Adding printers from AOS/VS II
 systems
 gathering information, 5-9
 testing the remote shell, 5-9
 using sysadm, 5-9
Adding printers from other vendors,
 procedures for, 4-1

- Adding printers on trusted systems, lpNet, 4-18
- Adding printers to /dev/lp, 3-2
 - example of, 3-5, 3-14
 - gathering printer information, 3-2
 - line printer mode, 3-4
 - lp() line controller, 3-3
 - procedures for, 3-2, 3-4, 3-14, 4-18
 - sysadm, 3-2, 3-14, 4-18
- Adding printers to datability termservers
 - example of, 6-6
 - hardware flow control, 6-5
 - print-ready mode, 6-5
 - sysadm, 6-6
- Adding printers to Duart ports, 3-7
 - example of, 3-8
 - gathering information, 3-7
 - print-ready mode, 3-8
 - procedures for, 3-7
- Adding printers to FastPort print servers, gathering information, 6-12
- Adding printers to MiLAN print servers
 - assigning IP addresses, 6-12
 - gathering information, 6-12
 - on trusted systems, 6-16
 - print-ready mode, 6-13, 6-16
 - sysadm, 6-13, 6-16
 - using sysadm, 6-12
- Adding printers to Syac controllers, 3-11
 - example of, 3-12
 - gathering information, 3-10
 - print-ready mode, 3-12
 - procedures for, 3-10
 - sysadm, 3-12
 - system file, 3-11
- Adding printers to termservers
 - defining termserver for DG/UX, 6-2
 - example of, 6-4
 - gathering information, 6-2
 - sysadm, 6-4
 - using sysadm, 6-2
- Adding remote BSD printers
 - example of, 4-12, 4-16
 - sysadm, 4-16
- Adding remote lpNet printers, example of, 4-4
- Adding remote lpNet printers on trusted systems, example of, 4-18
- Adding remote printers
 - attached to 5.4 systems, 4-2, 4-17, 7-17
 - attached to a network, 7-20
 - attached to BSD systems, 5-1, 5-14
 - attached to lpd systems, 4-2, 4-17, 7-17
 - attached to systems prior to DG/UX 5.4, 5-1, 5-14, 7-18
 - from AOS/VS systems, 5-7
 - gathering information, 4-3, 4-11, 4-15, 5-3, 5-10
 - NetWare, 4-11, 4-12
 - Novell, 4-12
 - on trusted systems, 5-14
 - registering hosts on the local system, 4-2
 - SunOS, 4-15
 - sysadm, 4-3, 5-11
 - system name, 8-8
- Adding remshlp printers
 - example of, 5-4, 5-15
 - print-ready mode, 5-4, 5-15
 - sysadm, 5-4, 5-15
- Administering the LP print service, suggestions for, 8-3
- Alert interval, 7-16
- Alert messages, 7-15
 - mail, 7-15
 - none, 7-16
 - quiet, 7-16
 - write, 7-16
 - writing custom scripts, 7-16
- Allowing access to forms, rules for, 9-6
- Allowing user access to print servers, .rhosts file, 5-2
- Analyzing data, troubleshooting, A-12
- ANNEX termservers
 - asynchronous cable for, 2-20
 - software flow control cable for, 2-20, 2-21
- ASCII format, 1-11
- Assigning IP addresses, sysadm, 6-8, 6-13
- Asynchronous port, B-1
- Automatic mode switching
 - printers, 1-9
 - sysadm, 1-10
 - using, 1-10

Autopush command, listing modules
pushed on /dev/lp, A-28

B

Banner pages, 8-25
Bannerid number, 9-64
components, 9-63
disabling, 8-26
interface program, 8-10
on trusted systems, 9-63
preventing spoofing, 9-64
reprinting, 9-63
requesting, 8-25
with multiple files, 9-66

Bannerid number, 9-64

Baud rate, 7-14, A-2

Berkeley print spooler. *See* Line Printer Daemon

Boot disk devices, default names of, B-13

Boot tape devices, default names of, B-13

Booting system diagnostics
from disk, B-12
from tape, B-13

Break points, troubleshooting, A-12

BSD printing, A-50
example of, A-53
lpc program, A-52
lpd program, A-53
printcap file, A-51
programs required, A-51
setting up for, A-51

C

Cables
asynchronous adapter, 2-3
connecting printers to RS-232 hosts, 2-16
connecting printers to VAC controllers, 2-16
connecting printers to VDA controllers, 2-16
connecting printers to VDC controllers, 2-16
for VTC boards, 2-11
list of, 2-15

Model 1340 EIA RS-232C, 2-18
Model 15275 Termserver/Modem, 2-19
Model 15293E030 Centronics, 2-21
Model 15307E RS-232C, 2-19
Model 15339E EIA RS-232C, 2-18
Model 15345E Centronics, 2-21
serial printers, 2-17

Canceling a print request, 7-36

Cat command, troubleshooting lpsched, A-8

CD-ROM devices, default names of, B-13

Character sets, 8-16
determining names of, 8-17
translating, 9-58

Cif input type, 7-11

Classes, 7-23

Cluster controller type, B-14

Cluster controllers, tty lines allocated to, B-22

Commands, format conventions, v

Configuring Datability termserver, example of, 2-9

Configuring DG/UX systems, for printing to Netware systems, 4-8

Configuring local printers
/dev/lp, 3-1
/dev/tty, 3-1
Duart port, 3-1

Configuring lpNet, for use with LPD, 4-10

Configuring Novell systems
FLex/IP, 4-7
for remote printing, 4-7
NetWare Loadable Modules (NLM), 4-7
NetWare NFS, 4-7
TCP/IP, 4-7

Configuring printers
accessed through the VTC, 2-11
commonly defined attributes, 8-4
components, 1-1
concepts, 1-1
connected to /dev/lp, 3-2
connected to BSD systems, 5-1
connected via remshlp, 5-1
connection method, 1-2
from shell, 8-1, 8-4
network connections, 1-4

- Configuring printers (*continued*)
 - on the network, 6-1
 - overview of, 4-1
 - sites, 8-3
 - software parameters, 1-4
 - Configuring remote printers, 4-1
 - Configuring SunOS systems
 - for remote printing, 4-13
 - procedures for, 4-14
 - Configuring TermServer 2000/2100, 2-7
 - example of, 2-7
 - Configuring the DG/UX system, for printing to SunOS systems, 4-14
 - Connecting parallel printers, VDA
 - cluster boxes, 2-16
 - Connecting printers, 8-6
 - cables required for RS-232 hosts, 2-16
 - directly to computers, 8-7
 - example for termservers, 2-7
 - FastPort print servers and the Ethernet, 2-13
 - hardware options for VTC boards, 2-10
 - installing hardware for VTC boards, 2-11
 - installing termserver hardware, 2-11
 - installing termserver software, 2-11
 - interface cables, 2-14
 - methods of, 8-6
 - procedures for VTC boards, 2-11
 - required cables for VAC controllers, 2-16
 - required cables for VDC controllers, 2-16
 - setting internet addresses for VTC boards, 2-11
 - software options for VTC boards, 2-10
 - to Duart ports, 2-3
 - to FastPort print servers, 2-5, 2-13
 - to hosts, 2-1, 2-3
 - to network connections, 2-5
 - to network devices, 2-1, 2-5
 - to TermServer 2000/2100, 2-7
 - to termservers, 2-5
 - to VAC controllers, 2-4
 - to VDA controllers, 2-4
 - to VTC boards, 2-5, 2-10, 2-11
 - Connecting printers to MiLAN print servers, cables required to, 2-13
 - Connection method, 8-6
 - Connection types
 - definition of, 3-3, 4-4, 5-4, 6-2, 7-4
 - list of, 7-4
 - Contacting Data General, vii
 - Content type, 1-5
 - definition of, 1-8
 - Content types, 8-11
 - list of, 8-12
 - simple, 8-13
 - specifying, 8-12
 - Controllers
 - order of device names, B-10
 - terminal line, B-10
 - Controlling access to printers, 8-24
 - Converting files, using filters, 9-10
 - Creating, printer class, 7-23
 - Curses language
 - description of, A-21
 - example of, A-21
 - Customizing the print service, 9-46
 - adjusting port characteristics, 9-48
 - adjusting the terminfo database, 9-49
 - methods of, 9-46
 - translating character sets, 9-58
 - writing a filter, 9-57
 - writing an interface program, 9-52
 - Cvr parameter, micro-stepping, A-21
- ## D
- Data Communications Equipment (DCE), definition of, 2-1
 - Data flow, troubleshooting, A-12
 - Data flow isolation techniques (DFI)
 - analyzing data, A-12
 - description of, A-12
 - determining data flow, A-12
 - example of, A-13, A-17
 - locating break points, A-12
 - trapping data, A-12
 - troubleshooting, A-1, A-12
 - troubleshooting local printers, A-13
 - troubleshooting remote printers, A-17
 - Data General, contacting, vii
 - Data Terminal Equipment (DTE), definition of, 2-1
 - Datability
 - interface script, 2-6, 6-1
 - termserver, 6-1

- Datability termserver, 2-9
 - adding printers to, 6-5
 - associating IP address with a port, 2-9
 - associating IP address with multiple ports, 2-9
 - creating an IP address for, 2-9
 - dedicated port, A-40
 - recommended parameters setting for, 2-9
 - setting port characteristics, 2-9
 - setting up, 2-9
- Dedicated port, datability termserver, A-40
- Default boot disk devices, table of, B-13
- Default boot tape devices, table of, B-13
- Default content type, simple, 8-13
- Default input type, simple, 8-13
- Default local printer attributes, 3-1
- Default printer attributes
 - discussion of, 8-26
 - list of, 3-2, 4-6, 5-6, 5-14, 6-11, 7-6, 7-7, 7-9, 7-10, 8-15
 - printers on network, 6-10
 - remote remsh on AOS/VS II, 5-13
 - remote lpNet, 4-6
 - remote remsh, 5-6
 - setting, 8-27
- Default printer destination, 8-28
 - specifying a, 8-28
- Default printer form attributes, list of, 9-5
- Default system disks, table of, B-13
- Defining filters, 9-14
- Defining termserver
 - /etc/hosts, 6-3
 - Internet address, 6-3
 - procedure for, 6-3
 - sysadm, 6-3
- Deleting
 - printer classes, 7-24
 - printer devices, 7-21
 - printer filters, 7-28
 - printer forms, 7-33
 - remote printers, 7-39
- Deleting filters, 9-14
- Detecting printer faults, using filters, 9-12
- Determining data flow, troubleshooting, A-12
- Device file, 3-3, 3-7, 3-10
- Device names, 6-3, 7-6
 - networked, 7-10
- Devices
 - adding printer devices, 7-3, 8-4
 - deleting printer devices, 7-21
 - displaying printer devices, 7-22
 - enabling and disabling printers, 7-22
 - modifying printer devices, 7-21
 - printers, 7-2
- DG 2x00 TermServers
 - assigning an IP address to a port, A-40
 - PVC, A-39
- DG/UX system, shutting down, B-11
- dg_annex, interface script, 6-1
- Diagnostic mode
 - accessing, A-34
 - MiLAN print servers, A-34
 - terminating, A-36
- Dial out failures, UUCP, A-5
- disable command, 8-34
 - example using the, 9-26
- Disabling printers, 7-22, 8-33
 - example of, 8-34
- Disabling software flow control, 2-4
- Disk, booting system diagnostics from, B-12
- Displaying
 - printer classes, 7-25
 - printer devices, 7-22
 - printer filters, 7-29
 - printer forms, 7-34
 - printer requests, 7-37
 - remote printer systems, 7-39
 - service status, 7-39
- Distributed printing configurations, 8-3
- Document types, 1-10
 - definition of, 1-10
 - print-ready state, 1-10
- Duart line controller, 3-7
- Duart ports, 3-7
 - adding printers to, 3-7
 - baud rate on, 3-7
 - connecting printers to, 2-3
 - hardware flow control, 2-3, 3-7
 - parity on, 3-7
 - serial connections, 1-3
 - software flow control, 3-7

E

- Emulation mode, 1-1, 1-5
- enable command, 8-34
 - example using the, 9-26
- Enabling hardware flow control, 2-4
 - ctsxon, 2-4
- Enabling printers, 7-22, 8-33
 - example of, 8-34
- Enabling remote printer access, on BSD systems, 5-1, 5-14
- Epsonfx input type, 7-12
- Establishing network connections
 - between host and printer port, A-37
 - Ethernet controllers, A-37
 - on term servers, A-38
 - VTC. *See* VME-bus Terminal Controller
- Ethernet controllers, establishing network connections, A-37
- Examining a printer configuration, 8-32
 - example of, 8-32
 - lpstat command, 8-32
- Examining filtered files, troubleshooting, A-16
- Examining filters, 9-23
- Examining forms
 - lpforms command, 9-9
 - lpstat command, 9-9

F

- FastPort print servers, 6-1
 - adding printers to the DG/UX system, 2-14
 - brief description of, 6-11
 - connecting printers to, 2-13
 - connecting to the Ethernet, 2-13
 - entering names into host tables, 2-14
 - installing on the network, 2-13
 - interface script, 6-1
 - local ports, A-33
 - parallel ports, 1-2, A-33
 - remote connection types, 1-3
 - self-test problems, A-34
 - serial connections, 1-3
 - serial port, A-33
 - setting up rarpd, 2-13
 - troubleshooting, A-1

- Fault detection, choosing an alert method, 8-22

- Fault recovery, 7-15, 8-23
 - beginning, 7-15
 - continue, 7-15
 - wait, 7-15

- Filter characteristics

- command, 9-14
- filter type, 9-16
- input types, 9-14
- list of, 9-14
- options, 9-16
- output types, 9-15
- printer types, 9-15
- printers, 9-15

- FILTER.NLM, definition of, 4-7

- filter.table.proto file

- entries defined, A-44
- example of, A-45
- filter name, A-45
- filter type, A-45
- how lpsched uses the, A-48
- input to filter, A-44
- options, A-45
- output from filter, A-45
- pathname to filter, A-45
- printer names, A-44
- printer type, A-44

- Filtered files

- examining, A-14
- troubleshooting, A-14

- Filters, 7-25, 9-9

- adding, 7-26, 9-14, 9-22
- characteristics of, 9-14
- converting files, 9-10
- default characteristics, 9-17
- defining, 7-25, 9-14
- defining options with templates, 9-18
- deleting, 7-28, 9-14
- detecting printer faults, 9-12
- displaying, 7-29
- evaluating programs for use as, 9-13
- examining, 9-23
- example of, 9-11, 9-20, 9-21, 9-22
- fast, 9-57
- for Postscript printers, 9-37
- guidelines for good, 9-13
- handling special modes, 9-11
- how to write, 9-57
- modifying, 7-28
- option keywords, 9-19
- purposes for, 9-10

- Filters** (*continued*)
- removing, 9-23
 - requirements of, 9-57
 - restoring, 7-28, 9-14
 - restoring factory defaults, 9-23
 - slow, 9-57
 - supplying definitions for, 9-17
 - translating character sets, 9-58
 - using for troubleshooting, A-15
- FleX/IP**, 4-7
- Flow control**
- ctsxon, 2-4
 - disabling, 2-4
 - hardware, 1-4, 2-3
 - on term servers, 6-2
 - serial connections, 1-4
 - Syac controllers, 3-10
 - XOFF, 2-3
 - XON, 2-3
- FLTRLIB.NLM**, definition of, 4-7
- Format conventions**, v
- Forms**, 7-29, 8-29
- adding, 7-30
 - allow list, 9-6
 - allow list of, 8-21
 - changing, 9-5
 - controlling the use of, 8-20
 - deleting, 7-33
 - deny list, 9-6
 - deny list of, 8-21
 - displaying, 7-34
 - examining, 9-9
 - lpforms command, 9-7, 9-8
 - modifying, 7-33
 - mounting, 7-33, 8-29
 - notification to mount, 9-7
 - removing, 9-5
 - restricting use of, 9-6
 - troubleshooting, A-1, A-20
 - unmounting, 7-33, 8-30
- Forms characteristics**
- character set choice, 9-3
 - comment, 9-3
 - defining characteristics of, 9-2
 - line pitch, 9-3
 - managing use of, 9-2
 - number of pages, 9-3
 - page length, 9-3
 - page width, 9-3
 - ribbon color, 9-3
- Forms characteristics, character pitch**, 9-3
- Fortram input type**, 7-12
- Full add feature**, definition of, 3-1, 7-10
- Fully qualified system names**, definition of, 8-8
- ## G
- Gathering printer information**
- connection types, 3-3
 - device file, 3-3
 - for /dev/lp, 3-2
 - for Duart ports, 3-7
 - for HP LaserJet 4 printers, 6-7
 - for printers attached to term servers, 6-2
 - for printers connected to FastPort print servers, 6-12
 - for printers connected to MiLAN print servers, 6-12
 - for remote printers attached to AOS/VS II systems, 5-10
 - for remote printers attached to pre-5.4 systems, 5-3
 - for remote printers connected to DG/UX systems, 4-3
 - for remote printers connected to LPD systems, 4-3
 - for remote printers connected to NetWare systems, 4-11
 - for remote printers connected to SunOS systems, 4-15
 - for Syac ports, 3-10
 - printer names, 3-2
 - printer types, 3-3
- ## H
- Handling special modes, with filters**, 9-11
- Hardware flow control**, 3-10
- ctsxon, 2-4
 - enabling, 2-4
 - on Datability term servers, 6-5
 - serial connections, 2-4
 - using, 2-3
 - wiring for, 2-3
- Hardware parameters**, 1-8
- specifying, 1-9
- Holding, print request**, 7-36

- Host names, 6-8, 6-12
 - Host tables, entering names for the FastPort, 2-14
 - Hosts, connecting printers to, 2-3
 - HP Laserjet 4 Printers, adding, 6-9
- I**
- iconv command, 9-58
 - iconv_data file, 9-58
 - Idle printers
 - busy dial-out ports, A-6
 - busy printers, A-6
 - filtering, A-5
 - printer fault, A-6
 - Illegible output, A-2
 - baud rate, A-2
 - default port settings, A-3
 - lpadmin command, A-3
 - parity setting, A-3
 - printer type, A-3
 - tabs, A-3
 - Initializing the printer, interface program, 8-10
 - Initializing the printer port, interface program, 8-10
 - Input type (content type), 1-8
 - Input types, 7-5, 7-8, 7-11, 8-11
 - cif, 7-11
 - definition of, 7-9
 - epsonfs, 7-12
 - fortram, 7-12
 - laserjet, 7-12
 - list of, 7-11, 8-12
 - otroff, 7-12
 - plot, 7-12
 - proprinter, 7-12
 - PS, 7-12
 - raster, 7-12
 - simple, 7-12, 8-13
 - specifying, 8-12
 - tex, 7-12
 - troff, 7-12
 - Installing PostScript fonts, 9-40
 - host-resident, 9-42
 - Installing PostScript printers, 9-38
 - Interconnectivity, general concepts of, 2-1
 - Interface cables
 - connecting printers to RS-232 hosts, 2-16
 - list of, 2-14, 2-15
 - Interface program, 8-10
 - definition of, 8-10
 - initialize the printer, 8-10
 - initializing the printer port, 8-10
 - manage printer faults, 8-10
 - print banner page, 8-10
 - run filters, 8-10
 - standard, 8-10
 - Interface programs
 - customizing, 9-55
 - how used, 9-53
 - list of tasks, 9-52
 - printer exit codes, 9-55
 - Interface script
 - termprinter_3com3.3, 2-6
 - termprinter_script, 2-6
 - Interface scripts
 - datability, 2-6
 - dg_annex, 6-1
 - list of, 7-12
 - network, 7-10
 - remote, 5-5, 5-16, 7-8
 - remshlp, 5-5, 5-16
 - remshlp_a, 5-5, 5-16
 - remshlp_bsd, 5-5, 5-16
 - termprinter_3com3.3, 6-1
 - termprinter_tcp, 2-6, 6-1
 - termprinter_telnet, 6-1
 - Internet addresses, lpsystem command, 8-9
 - IP addresses, 6-7, 6-12
 - Assigning to HP LaserJet 4 printer with HP JetDirect, 6-8
 - assigning to print servers, 6-13
- L**
- Labels, Least Upper Bound (LUB), 9-63
 - Laserjet input type, 7-12
 - ldterm
 - line discipline module, A-28
 - streams modules, A-28
 - Least Upper Bound (LUB), 9-63

- Line controller, terminal, B-1
- Line number, of port, 3-8, 3-11
- Line Printer Daemon (LPD), 4-1
- Line printer mode, adding printers to /dev/lp, 3-4
- Line printers, 1-1
- Lines per inch, micro-stepping, A-20
- Listing forms, methods of, 8-20
- Local printer attributes, default, 3-1
- Local printer names, 4-3, 4-11, 4-16, 5-3, 5-10
- Locating break points, troubleshooting, A-12
- Locating filtered files, example of, A-14
- Log file, LP print service, 9-32
- Log files
 - /var/lp/logs/lpNet, 9-32
 - /var/spool/lp/logs/lpsched, 9-32
 - /var/spool/lp/logs/requests, 9-33
- Logical disk pieces, B-12
- Login terminal, printers, 8-7
- lp () line controller, 3-3
- lp command, printing on AOS/VS systems, 5-7
- LP commands
 - list of, 8-1, 9-1
 - lpforms, 9-4
- LP components
 - hardware, 8-2
 - software, 8-2
- LP functions, 8-2
 - error tracking, 8-3
 - filtering user files, 8-3
 - keeping track of forms, 8-3
 - keeping track of job status, 8-3
 - keeping track of print wheels, 8-3
 - scheduling print requests, 8-2
 - scheduling work of multiple printers, 8-2
 - starting programs, 8-3
- LP print service
 - accept, 7-17
 - accepting requests, 7-22
 - access control, 8-24
 - adding classes, 7-23
 - adding devices, 7-3, 8-4
 - adding filters, 7-26
 - adding forms, 7-30
 - adding printers, 8-31
 - adding remote printers, 7-38
 - alert interval, 7-16
 - alert messages, 7-15
 - auditing, 9-67
 - banner pages, 8-25
 - baud settings, 7-14
 - canceling a request, 7-36
 - character sets, 8-16, 9-58
 - classes, 7-23, 8-27
 - command summary, 9-1
 - components of, 8-2
 - configuring, 8-4
 - configuring printer sites, 8-3
 - connection type, 7-4
 - content types, 8-11
 - cron jobs, 9-32
 - customizing, 9-46
 - default destination, 7-21, 8-28
 - deleting classes, 7-24
 - deleting devices, 7-21
 - deleting filters, 7-28
 - deleting forms, 7-33
 - deleting remote printers, 7-39
 - disabling printers, 7-22, 8-33
 - displaying classes, 7-25
 - displaying devices, 7-22
 - displaying filters, 7-29
 - displaying forms, 7-34
 - displaying remote systems, 7-39
 - displaying requests, 7-37
 - displaying service status, 7-39
 - distributed, 8-3
 - enable, 7-17
 - enabling printers, 7-22, 8-33
 - Epson, 7-12
 - fault detection, 8-21
 - fault recovery, 7-15
 - filters, 7-25, 9-9, 9-57
 - forms, 7-29, 8-20, 8-29, 9-2
 - forms to allow, 7-17
 - forms to deny, 7-17
 - Full Add operation, 7-10
 - functions of, 8-2
 - Hewlett-Packard LaserJet, 7-12
 - holding requests, 7-36
 - IBM ProPrinter, 7-12
 - input type, 7-11
 - input types, 8-11
 - interface program, 8-10, 9-52
 - interface script, 7-12
 - load management, 9-24
 - log file, 9-32

- LP print service (*continued*)
 - lp.proto crontab, 9-32
 - lpNet file, 9-32
 - lpsched file, 9-32
 - managing devices, 7-2
 - managing with sysadm, 7-1
 - menu and command summary, 8-1
 - modifying classes, 7-24
 - modifying devices, 7-21
 - modifying filters, 7-28
 - modifying forms, 7-33
 - modifying remote systems, 7-39
 - mounting forms, 7-33
 - moving requests, 7-36, 9-25
 - network configuration, 8-4
 - overview of, 8-2
 - port characteristics, 9-48
 - PostScript fonts, 9-40
 - PostScript printers, 9-35
 - print options, 7-14
 - print style, 8-26
 - print wheels, 8-16, 8-29
 - printer configuration, 8-32
 - printer description, 7-17
 - printer descriptions, 8-26
 - printer options, 7-14
 - printer ports characteristics, 8-14
 - printer types, 7-11, 8-11
 - queue priorities, 9-27
 - Quick Add operation, 7-5
 - recovering from faults, 8-23
 - rejecting requests, 7-22
 - remote connections, 8-8
 - remote printers, 7-37, 8-9
 - requests, 7-36, 9-25
 - requests file, 9-33
 - restoring filters, 7-28
 - resume a held request, 7-36
 - scheduler, 7-37
 - server configuration, 8-3
 - setting priorities, 7-34
 - shell management commands, 8-1
 - starting the, 9-30
 - stopping the, 9-30
 - stty options, 7-14
 - terminfo file, 9-49
 - tips for administering, 8-3
 - troubleshooting, A-1
 - unmounting forms, 7-33
 - users to allow, 7-16
 - users to deny, 7-16
- LP printer service, processing a print request, 9-47
- LP scheduler, troubleshooting, A-1
- LP subsystem, auditing, 9-67
- lp.proto file, 9-32
- lpadmin command
 - adding printers, 8-31
 - alerting to mount print wheels, 8-19
 - connection method, 8-7
 - controlling printer access, 8-24
 - creating printer classes, 8-28
 - fault detection, 8-22
 - fault recovery, 8-23
 - forms allow list, 8-21
 - forms deny list, 8-21
 - hardcopy terminal, 8-7
 - interface program, 8-11
 - PostScript printers, 9-38
 - printer device, 8-7
 - printer name, 8-5
 - removing a printer class, 8-31
 - setting default printer attributes, 8-27
 - setting printer port characteristics, 8-15
 - specifying character sets, 8-17
 - specifying content types, 8-12
 - specifying default printer destination, 8-28
 - specifying input types, 8-12
 - specifying print wheels, 8-17
 - specifying printer, 8-11
- Lpc program, discussion of, A-52
- LPD, configuring lpNet for, 4-10
- Lpd program, A-50
- lpfilter, restoring defaults, 9-23
- lpfilter command
 - adding filters, 9-14
 - adding filters to system, 9-22
 - defining filter characteristics, 9-17
 - defining options with templates, 9-18
 - deleting filters, 9-14
 - examining filters, 9-23
 - removing filters, 9-23
 - restoring filters, 9-14
- lpforms command
 - alerting to mount forms, 9-8
 - allowing access to forms, 9-7
 - defining forms, 9-4
 - denying access to forms, 9-7
 - examining forms, 9-9
 - removing a form, 9-6

- lpmove command
 - example using the, 9-26, 9-27
 - moving print requests, 9-26
 - moving requests, 9-25
 - lpNet, configuring to use LPD, 4-10
 - lpNet daemon, 4-1
 - lpNet file, 9-32
 - lpsched, printing on remote machines, A-49
 - lpsched command, starting the LP service, 9-31
 - lpsched file, 9-32
 - lpshut command, stopping the LP service, 9-31
 - lpstat command
 - examining forms, 9-9
 - examining printer configurations, 8-32
 - lpsystem command
 - adding a system entry, 8-9
 - getting Internet addresses, 8-9
 - making remote systems available, 8-8
 - lpusers command
 - assigning queue priority limits, 9-27
 - examining queue priorities, 9-29
 - example using the, 9-28
 - LUB (Least Upper Bound) label, 9-63
- M**
- Maintaining PostScript fonts, 9-40
 - Maintaining PostScript printers, 9-38
 - filters, 9-39
 - Manage printer faults, interface script, 8-10
 - Managing
 - PostScript printers, 9-35
 - print load, 9-24
 - printer classes, 7-23
 - printer devices, 7-2
 - printer filters, 7-25
 - printer forms, 7-29
 - printer requests, 7-36
 - printers on the network, 7-37
 - remote printers, 7-37
 - Managing PostScript fonts
 - downloading host-resident, 9-44
 - host-resident, 9-42
 - Managing queue priorities, 9-27
 - default priority, 9-27
 - default priority limit, 9-27
 - priority limit, 9-27
 - Managing the LP service, with sysadm, 7-1
 - Managing the printing load, 9-24
 - accept command, 9-24
 - lpmove command, 9-24
 - reject command, 9-24
 - Mandatory access control (MAC), Least Upper Bound (LUB) label, 9-63
 - Matching printer type to printer, terminfo database, A-8
 - Micro-stepping
 - calculating lines per inch, A-20, A-21
 - cvr parameter, A-21
 - example of, A-21
 - orvi parameter, A-21
 - vertical spacing, A-20
 - MiLAN print servers
 - accessing diagnostic mode, A-34
 - adding printers to, 6-13, 6-16
 - brief description of, 6-11
 - determining Ethernet cable in use, A-37
 - diagnostic mode, A-34
 - local ports, A-33
 - parallel ports, 1-2, A-33
 - printers connected to, 6-11
 - remote connection types, 1-3
 - self-test problems, A-34
 - serial connections, 1-3
 - serial port, A-33
 - terminating diagnostic mode, A-36
 - troubleshooting, A-1, A-33, A-35
 - Modifying
 - printer classes, 7-24
 - printer devices, 7-21
 - printer filters, 7-28
 - printer forms, 7-33
 - remote systems, 7-39
 - Mounting a form
 - example of, 8-30
 - procedure for, 8-29
 - Mounting a print wheel
 - alerting, 8-18
 - example of, 8-30
 - procedure for, 8-29

Mounting printer forms, 7-33, 8-29
 alerting, 9-7

Moving print requests, 7-36

N

Naming printers, 3-2, 3-7, 4-3, 4-11, 5-3,
 5-10, 6-2, 6-8, 6-12, 8-5
 example of, 3-4, 4-4, 5-4, 5-11, 6-4

NetWare Loadable Modules (NLM), 4-7
 FILTER.NLM, 4-7
 FLTRLIB.NLM, 4-7
 PLPD.NLM, 4-7
 PLPDCFG.NLM, 4-7

NetWare NFS, 4-7

NetWare printer
 allowing host access, 4-8
 sharing resources, 4-8

NetWare printer resources, allowing
 access to, 4-8

NetWare systems, configuring for
 remote printing, 4-7

Network configurations, example of, 8-4

Network connections, 1-4
 advantages of, 1-4
 connecting printers to, 2-5
 disadvantages of, 1-4

Network installation, for FastPort print
 servers, 2-13

Networking problems
 conflicting accept/reject messages, A-7
 jobs backing up in local queue, A-6
 jobs backing up in remote queue, A-7

New Line key, v

NLM. *See* NetWare Loadable Module

Notational conventions in this manual,
 v

Novell systems, 4-6
See also NetWare systems

O

Orvi parameter, micro-stepping, A-21

Otroff input type, 7-12

Overview, LP print service, 8-2

P

Page printers, 1-1

Parallel connections, 1-2
 advantages of, 1-2
 setting up, 1-2

Parallel ports
 direct connection to, 1-2
 network connections, 1-2
 VDA cluster box, 1-2

Parallel printer cable, 2-21

Parallel printers
 flow control for, 2-4
 Model 15293E030 Centronics cable,
 2-21
 Model 15345E Centronics cable, 2-21
 troubleshooting notes, A-30
 verifying connection and printer
 setup, A-8

Parallel printing, /dev/lp port, 3-2

Parity, setting, A-3

Permanent Virtual Circuit (PVC), 2-12,
 A-39
 setting the, 2-12

Pieces of logical disk, B-12

Pin diagrams
 asynchronous 4-wire cable, 2-5
 asynchronous 5-wire cable for
 ANNEX termservers, 2-20, 2-21
 asynchronous adapter cable, 2-3
 asynchronous cable for TermServer
 2000/2100, 2-6, 2-7, 2-10
 asynchronous DTE cable for ANNEX
 termservers, 2-20
 Model 1340 EIA RS-232C, 2-18
 Model 15275 Termserver/Modem
 cable, 2-19
 Model 15307E RS-232C cable, 2-19
 Model 15339E EIA RS-232C cable,
 2-18
 Model 15426E asynchronous cable,
 2-17

Plot input type, 7-12

PLPD.NLM, definition of, 4-7

PLPDCFG.NLM, definition of, 4-7

pmadm command, configuring the port
 monitor, 8-9

Port characteristics, printers, 8-14

Port monitors, pmadm, 8-9

- Port numbers, 6-3, 6-8, 6-12, 7-10
- Ports, determining line number of, 3-8, 3-11, B-1, B-20
- Post processing, problems with, A-29
- Postprint filter, data flow isolation techniques, A-13
- PostScript filters, list of, 9-39
- PostScript fonts, 9-40
 - host-resident, 9-42, 9-44
 - installing, 9-40
 - maintaining, 9-40
 - permanently downloaded, 9-45
 - printer-resident, 9-41
- PostScript formatting language, 1-11
 - example of, 1-12
- PostScript printers
 - capabilities provided filters, 9-37
 - installing, 9-35, 9-38
 - lp command, 9-36
 - lpadmin command, 9-38
 - maintaining filters, 9-39
 - managing, 9-35
 - optional filters, 9-37
 - supporting non-PostScript requests, 9-36
 - using, 9-36
- PREDITOR utility, 5-7
- Print classes, discussion of, 8-27
- Print filters
 - filter.table.proto file, A-44
 - troubleshooting, A-1
- Print options
 - banner, 7-15
 - banneroff, 7-15
 - characters per inch (dpi), 7-14
 - length, 7-14
 - lines per inch (lpi), 7-14
 - list of, 7-14
 - nobanner, 7-15
 - width, 7-14
- Print requests
 - accepting, 8-33
 - rejecting, 8-33
- Print scheduler
 - starting the, 9-30
 - stopping the, 9-30
- Print server configurations, 8-3
 - example of, 8-4
- Print servers on BSD systems, allowing user access to, 5-2
- Print wheels, 8-16, 8-29
 - alerting to mount, 8-18
 - mounting, 8-29
 - unmounting, 8-30
- Print-ready format, 1-11
 - PostScript formatting language, 1-11
- Print-ready mode
 - opost option, 6-2
 - adding printers to /dev/lp, 3-5, 3-14
 - adding printers to Duart ports, 3-8
 - adding printers to Syac controllers, 3-12
 - Duart ports, 3-8
- Printcap entry, explanation of, A-52
- Printcap file, example entry, A-52
- Printer
 - adding a, B-3
 - line number of a, 3-11
- Printer access
 - allow list, 8-24
 - deny list, 8-24
 - on a trusted system, 9-66
- Printer access on BSD systems, enabling, 5-1, 5-14
- Printer access on DG/UX systems prior to 5.4, enabling, 5-1, 5-14
- Printer addresses, 6-2
- Printer attributes
 - commonly defined, 8-4
 - connection method, 8-6
 - content types, 8-11
 - default, 3-1, 6-10, 8-26
 - default remsh, 5-6
 - default remsh on AOS/VS II, 5-13
 - defaults for remote lpNet, 4-6
 - input types, 8-11
 - interface program, 8-10
 - printer name, 8-5
 - printer types, 8-11
 - setting default, 8-27
 - specifying, 8-27
- Printer characteristics
 - alignment pattern, 9-4
 - emulation mode, 1-1
 - line printers, 1-1
 - page printers, 1-1
- Printer classes
 - names, 8-28
 - removing, 8-30

- Printer concepts, 1-1
 - configuring printers, 1-1
 - printer characteristics, 1-1
- Printer configurations, examining, 8-32
- Printer descriptions, 8-26
 - adding, 8-26
- Printer devices, troubleshooting, A-1
- Printer exit codes, list of, 9-56
- Printer fault recovery, 8-23
- Printer faults
 - alert messages, 7-15
 - alerts to, 8-21
 - methods to alert to, 8-22
 - recovering from, 8-23
- Printer information
 - connection types, 3-7, 3-10, 4-4, 4-11, 4-16, 5-4, 5-10, 6-2, 6-8, 6-12
 - device file, 3-7, 3-10
 - device names, 6-3
 - host names, 6-8, 6-12
 - IP addresses, 6-7, 6-12
 - local printer names, 4-3, 4-11, 4-16, 5-3, 5-10
 - port numbers, 6-3, 6-8, 6-12
 - printer addresses, 6-2
 - printer names, 3-7, 3-10, 6-2, 6-8, 6-12
 - printer types, 3-7, 3-11, 4-4, 4-11, 4-16, 5-4, 5-10, 6-3, 6-8, 6-12
 - remote host names, 4-3, 4-11, 4-15, 5-3, 5-10
 - remote printer names, 4-3, 4-15, 5-3, 5-10
- Printer load, managing, 9-24
- Printer modes, switching, 1-9
- Printer name, 8-5
- Printer names, 3-2, 3-7, 3-10, 6-2, 6-8, 6-12, 7-4
- Printer port characteristics
 - default, 8-14
 - list of, 8-14
 - specifying additional, 8-15
 - stty command, 8-14
- Printer queues, moving requests, 9-29
- Printer specific terminfo entries, list of, 1-9
- Printer types, 1-5, 3-3, 3-7, 3-11, 4-16, 5-4, 5-10, 6-3, 6-8, 6-12, 7-5, 7-7, 7-9, 7-11, 8-11
 - definition of, 1-5
 - description of, 1-5
 - list of, 1-5
 - printer specific entries, 1-9
 - specifying, 8-11
 - terminfo database, 4-11
- Printer worksheets, C-1
 - list of, C-1
- Printers
 - adding remote BSD, 4-11, 4-15
 - configuring from shell, 8-4
 - configuring sites, 8-3
 - connected to Duart ports, 3-7
 - connected to MiLAN print servers, 6-11
 - connected to termservers, 6-2
 - controlling access to, 8-24
 - determining where to place, 8-3
 - faults, 8-21
 - line number of, 3-8
 - port characteristics, 8-14
 - removing, 8-30
 - sharing resources, 4-2
 - using as a login terminal, 8-7
 - with network adapter cards, 6-1
 - worksheets for, C-1
- Printers accessed through the VTC, setting up, 2-11
- Printers connected to Syac controllers, 3-10
- Printers on the network
 - MiLAN FastPort print server, 6-1
 - termservers, 6-1
- Printing configurations
 - direct, 8-3
 - print server, 8-3
- Printing files
 - banner pages, 9-63
 - canceling a print request, 9-63
 - components of banner pages, 9-63
 - preventing banner page spoofing, 9-64
 - when a file is rejected, 9-63
- Printing on AOS/VS systems
 - lp command, 5-7
 - rlp command, 5-7
- Printing requests
 - accepting, 9-25
 - holding, 9-29

- Printing requests (*continued*)
 - moving, 9-25, 9-29
 - processing, 9-47
 - rejecting, 9-24
 - Printing to NetWare systems,
 - configuring DG/UX systems, 4-8
 - Printing with BSD, A-50
 - example of, A-53
 - printcap file, A-51
 - programs required, A-51
 - setting up for, A-51
 - troubleshooting, A-1
 - Proprietary input type, 7-12
 - Prototype files, lp.proto, 9-32
 - PS input type, 7-12
 - ptem
 - pseudo terminal emulator module, A-30
 - streams modules, A-30
 - PVC, A-39
 - See also* Permanent Virtual Circuit
 - DG 2x00 TermServer, A-39
- Q**
- Queue priorities
 - changing, 9-29
 - default priority, 9-27
 - default priority limit, 9-27
 - examining, 9-29
 - holding requests, 9-29
 - managing, 9-27
 - moving requests, 9-30
 - priority limit, 9-27
 - setting default, 9-28
 - setting limits, 9-28
 - Quick add feature, definition of, 3-1, 7-5
- R**
- Rarpd server, setting up for FastPort print servers, 2-13
 - Raster input type, 7-12
 - Recovering from printer faults, 8-23
 - Registering hosts on the local system, sysadm, 4-2
 - Registering local host on remote systems, sysadm, 4-2
 - Registering local system on remote systems, example of, 4-2
 - Registering NetWare machines
 - on local DG/UX systems, 4-9
 - sysadm, 4-9
 - Registering SunOS systems
 - on your local system, 4-14
 - sysadm, 4-14
 - reject command
 - example using the, 9-27
 - rejecting print requests, 8-33, 9-25
 - Rejecting print requests, 8-33
 - example of, 8-33
 - Related Data General manuals, iv
 - Related other documents, v
 - Remote host names, 4-3, 4-11, 4-15, 5-3, 5-10, 7-8
 - Remote lpNet connections, 8-8
 - Remote lpNet printers
 - adding, 4-4
 - adding on trusted systems, 4-18
 - Remote print servers, allowing host access, 5-2
 - Remote printer access
 - enabling on BSD systems, 5-1
 - enabling on pre-5.4 DG/UX systems, 5-1
 - on DG/UX system prior to 5.4, 5-1
 - on DG/UX systems prior to 5.4, 5-14
 - Remote printer names, 4-3, 4-15, 5-3, 5-10, 7-8
 - Remote printer resources, allowing access to, 4-2
 - Remote printers
 - ./rhosts file, 5-2
 - /etc/hosts.equiv, 5-2
 - adding, 4-3
 - lpnet, 4-4, 4-18
 - setting up AOS/VS systems for, 5-7
 - Remote printing
 - accessing hosts, 4-2
 - allowing host access, 4-14, 5-2, 8-9
 - configuring the local port monitor, 8-9
 - on AOS/VS II systems, 5-7
 - PREDITOR utility, 5-7
 - troubleshooting, A-1, A-49
 - with lpsched, A-49

- Remote shell, testing the, 5-10
 - Removing
 - printer classes, 7-24
 - printer devices, 7-21
 - printer filters, 7-28
 - printer forms, 7-33
 - remote printers, 7-39
 - Removing a form, example of, 9-6
 - Removing a print class, lpadmin command, 8-31
 - Removing filters, 9-23
 - example of, 9-23
 - Remshlp, 5-1
 - Remshlp interface script, description of, 5-5, 5-16
 - Remshlp_a, description of, 5-5, 5-16
 - Remshlp_bsd, description of, 5-5, 5-16
 - requests file, 9-33
 - Restoring, printer filters, 7-28
 - Restoring filters, 9-14
 - Resuming, a held printer request, 7-36
 - Rlp command, printing on AOS/VS systems, 5-7
 - RS-232/422 ports
 - sample worksheet for, B-6
 - worksheet for, B-6, C-5
 - RS-232C pin definitions, 2-2
 - RS-232C standard, Data Terminal Equipment (DTE), 2-1
 - RS232C standard, Data Communications Equipment (DCE), 2-1
 - Running filters, interface program, 8-10
- S**
- Scheduler, starting and stopping the, 7-37
 - SCM, B-12
 - Serial connections, 1-2
 - advantages of, 1-2, 1-4
 - flow control, 1-4
 - hardware flow control, 1-4
 - setting up, 1-3
 - sysadm, 1-3
 - termservers, 6-2
 - Serial printers
 - asynchronous adapter cable, 2-3
 - cables for, 2-17
 - connected to Duart ports, 3-7
 - Model 1340 EIA RS-232C cable, 2-18
 - Model 15275 Termserver/Modem cable, 2-19
 - Model 15307E RS-232C cable, 2-19
 - Model 15339E EIA RS-232C cable, 2-18
 - Model 15426E interface cable, 2-17
 - troubleshooting, A-8
 - setd command, setting up printers accessed through the VTC, 2-12
 - Setting, printer priorities, 7-34
 - Setting up Datability termservers, 2-9
 - Setting up BSD printers, example of, A-53
 - Setting up Datability termservers, 2-9
 - example of, 2-9
 - Setting up parallel connections, sysadm, 1-3
 - Setting up parallel ports, 1-2
 - on a FastPort print server, 1-2
 - on a VDA cluster box, 1-2
 - stty options, 1-3
 - Setting up printers
 - accessed through the VTC, 2-11
 - connected to /dev/lp, 1-2
 - Setting up printers accessed through the VTC, Permanent Virtual Circuit (PVC), 2-12
 - Setting up queues, 1-4
 - Setting up serial connections, 1-3
 - Duart port, 1-3
 - FastPort print server, 1-3
 - MiLAN print servers, 1-3
 - termservers, 1-3
 - VAC 16 cluster box, 1-3
 - VDA cluster box, 1-3
 - Setting up TermServer 2000/2100, 2-7
 - Baud parameter, 2-7
 - example of, 2-7
 - FlowControlFrom parameter, 2-7
 - FlowControlTb parameter, 2-7
 - setd command, 2-8
 - Shell, configuring printers from, 8-1
 - Shutting down the system, B-11

- Simple input type, 7-12
- Simple system names, definition of, 8-8
- Software configurable items, 1-4
- Software flow control, 3-10
 - disabling, 2-4
- Software parameters, 1-5
 - content type, 1-8
 - emulation mode, 1-5
 - printer type, 1-5
 - stty options, 1-8
- Specifying content types
 - stty options, 1-8
 - sysadm, 1-8
- Specifying hardware parameters, 1-9
- Starting the LP service, lpsched
 - command, 9-31
- Starting the print scheduler, 7-37
- Stopping the LP service, lpshut
 - command, 9-31
- Stopping the print scheduler, 7-37
- Streams
 - definition of, A-25
 - driver component, A-25
 - head component, A-25
 - modules components, A-25
 - transferring data through, A-26
 - troubleshooting, A-1
- Streams devices
 - /dev/lp, A-28
 - parallel port, A-27
 - serial lines, A-27
 - troubleshooting, A-1
- Streams driver, definition of, A-26
- Streams modules
 - definition of, A-25
 - ldterm, A-28
 - pem, A-28
 - pushing on the /dev/lp device, A-27
 - pushing on the stream, A-26
 - ttcompat, A-28
- stty command
 - printer port characteristics, 8-14
 - with printers, 7-14
- stty options, 1-5
 - disabling software flow control, 2-4
 - enabling hardware flow control, 2-4
 - parallel ports, 1-3
- SunOS printers
 - allowing access to, 4-14
 - sharing resources, 4-14
- SunOS systems, 4-13
 - allowing host access to, 4-14
 - configuring for remote printing, 4-13
 - registering on your local system, 4-14
- Supporting PostScript printers
 - administrator's duties, 9-38
 - filters, 9-37
- Switching printer modes
 - automatic, 1-9
 - PostScript language (PS), 1-9
 - Printer Control Language (PCL), 1-9
- Syac controllers, 3-11
 - adding printers to, 3-10
 - flow control on, 3-10
 - hardware flow control, 3-10
 - printers connected to, 3-10
 - software flow control, 3-10
- Sysadm
 - adding a remote 5.4 printer, 4-3
 - adding a remote BSD printer, 4-11, 4-12, 4-15, 4-16
 - Adding HP LaserJet 4 printers with HP JetDirect, 6-9
 - adding NetWare machines, 4-8
 - adding printers to /dev/lp, 3-2, 3-14, 4-18
 - adding printers to MiLAN print servers, 6-13, 6-16
 - adding printers to Syac controllers, 3-12
 - adding printers to termserver, 6-4
 - adding remote printers, 4-4, 4-19, 5-11
 - adding remshlp printers, 5-4, 5-15
 - allowing host access, 5-2
 - assigning IP addresses, 6-8, 6-13
 - defining termserver, 6-3
 - managing the LP service, 7-1
 - registering hosts on the local system, 4-2
 - registering local host on remote systems, 4-2
 - registering NetWare machines, 4-9
 - registering SunOS systems, 4-14
 - serial connections, 1-3
 - switching printer modes, 1-10
 - systems, 7-2
- Sysadm printer menus
 - Classes, 7-1
 - Devices, 7-1

- Sysadm printer menus (*continued*)
 - filters, 7-1
 - Forms, 7-2
 - List, 7-2
 - Priorities, 7-2
 - Requests, 7-2
 - Scheduler/Service, 7-2
 - Sysadm printer operations
 - accept, 7-3
 - add, 7-3
 - default, 7-3
 - delete, 7-3
 - disable, 7-3
 - enable, 7-3
 - list, 7-3
 - modify, 7-3
 - reject, 7-3
 - Systech contoller pin number mapping, 2-2
 - System configuration file, B-10
 - System diagnostics, B-11
 - booting from disk, B-12
 - booting from tape, B-13
 - System file
 - adding printers to Syac controllers, 3-11
 - excerpt from, 3-3, 3-4, B-10
 - System name, 8-8
- T**
- Tape, booting system diagnostics from, B-13
 - tload command, setting up printers accessed through the VTC, 2-12
 - TCP/IP, 4-7
 - Terminal line controller, B-1
 - system diagnostics, B-11
 - Terminal line controllers, tty lines allocated to, B-21
 - Terminal lines, sample worksheet for, B-4
 - Terminals
 - adding, B-1
 - line controller type, B-14
 - Terminfo, 1-5
 - checking entries with tput command, A-9
 - printer definitions, 1-5
 - troubleshooting, A-1, A-20
 - Terminfo database, 4-4, 4-11
 - curses language, A-20
 - Terminfo entries, printer specific, 1-9
 - terminfo file, 9-49
 - termprinter_3com3.3, interface script, 2-6, 6-1
 - termprinter_tcp, interface script, 2-6, 6-1
 - termprinter_telnet, interface script, 2-6, 6-1
 - TermServer 2000/2100, 2-7
 - asynchronous cable for, 2-7, 2-10
 - setd command, 2-8
 - software flow control cable for, 2-6
 - Termservers
 - adding printers to, 6-2
 - Annex Three Host-boot, 6-1
 - Annex Three Self-boot, 6-1
 - common problems and fixes, A-44
 - configuring printer ports, A-40
 - connecting printers to, 2-5
 - connecting serial printers to, 2-5
 - CS/2000, 6-1
 - CS/2100, 6-1
 - CS/3000, 6-1
 - CS/3100, 6-1
 - CS2000, 6-1
 - datability, 6-1
 - defining for DG/UX, 6-3
 - establishing network connections, 2-5
 - establishing network connections, A-38
 - flow control, 6-2
 - installing hardware for, 2-11
 - installing software for, 2-11
 - MicroAnnex Three self-boot, 6-1
 - MicroAnnex XL Host-boot, 6-1
 - printers connected to, 6-2
 - printing from an Ethernet controller, A-41
 - printing on, A-37
 - serial connections, 1-3, 6-2
 - troubleshooting, A-1, A-42
 - Tex input type, 7-12
 - tic command, 7-5
 - tput command
 - checking terminfo entries, A-9
 - determining names of character sets, 8-17
 - example of, A-10

- Trapping data
 - example of, A-13
 - troubleshooting, A-12
 - Troff input type, 7-12
 - Troubleshooting
 - analyzing data, A-12
 - baud rate, A-2
 - data flow, A-1
 - determining data flow, A-12
 - dial out failures, A-5
 - examining filtered files, A-16
 - example using data flow isolation techniques, A-13, A-17
 - idle printers, A-5
 - illegible output, A-2
 - locating break points, A-12
 - LP scheduler, A-1, A-7
 - MiLAN FastPort devices, A-20
 - MiLAN FastPort print servers, A-1, A-33
 - networking problems, A-6
 - no output, A-1
 - parallel printers, A-30
 - postprint filter, A-13
 - print filters, A-1, A-20
 - printing with BSD, A-1, A-20
 - remote printing, A-1, A-20
 - setting parity, A-3
 - streams and printers devices, A-1, A-20
 - terminfo and forms, A-1, A-20
 - term servers, A-1, A-20
 - trapping data, A-12
 - wrong characters printed, A-4
 - wrong spacing, A-3
 - Troubleshooting commands, MiLAN print servers, A-35
 - Troubleshooting lpsched
 - cat command, A-8
 - matching filters, document, and printer, A-8
 - matching interface script and printer, A-8
 - matching printer type and printer, A-7
 - procedures for, A-7
 - serial printers, A-8
 - standard interface script, A-10
 - verifying the connection and setup, A-7
 - Troubleshooting MiLAN print servers
 - Ethernet light, A-33
 - parallel light, A-34
 - power light, A-33
 - serial light, A-34
 - Troubleshooting procedures
 - for local printers, A-13
 - for parallel printers, A-30
 - for remote printers, A-17, A-50
 - for term servers, A-42
 - Trusted DG/UX
 - adding remote printers on, 4-18
 - remote lpnet printers, 4-18
 - Trusted systems
 - adding printers to MiLAN print servers, 6-16
 - adding remote printers on, 4-19
 - adding remote printers to, 5-14
 - ttcompat, streams modules, A-28
 - Tty lines
 - allocated to controller addresses, B-22
 - allocated to terminal line controllers, B-21
 - sample worksheet for, B-4
 - worksheet for, B-5, C-4
- U**
- Unmounting printer forms, 7-33
- V**
- VAC 16 cluster box, serial connections, 1-3
 - VAC 16 controllers, printers connected to, 3-10
 - VAC controller, gathering information for, B-1
 - VAC controllers
 - connecting parallel printers to, 2-4
 - connecting printers to, 2-4
 - connecting serial printers to, 2-4
 - VAC/16 controller
 - sample worksheet for, B-7
 - worksheet for, B-7, C-5
 - VDA cluster box
 - connecting parallel printers to, 2-16
 - parallel ports, 1-2
 - serial connections, 1-3

VDA controllers

- connecting parallel printers to, 2-4
- connecting printers to, 2-4
- connecting serial printers to, 2-4
- flow control on, 2-4
- gathering information for, B-1
- printers connected to, 3-10

VDA host adapter

- sample worksheet for, B-8
- worksheet for, B-9, C-6

VDA/VAC controllers, asynchronous

- 4-wire cable for, 2-5

VDC controllers, printers connected to, 3-10**Verifying connection and printer setup, parallel printers, A-8****Vertical spacing, micro-stepping, A-20****VME-bus Terminal Controller. *See* VTC****VTC boards**

- connecting printers to, 2-10
- determining cables for, 2-11
- establishing network connections, A-37
- installing, 2-11
- printing through, 2-11
- printing to termserver, A-38
- setting internet addresses, 2-11
- setting termserver hardware and software, 2-11

setting termserver ports for, 2-11

setting up hosts to print on, 2-11

vtc.addr file, A-38

VTC to termserver printing, example of, A-39

vtc.addr file, configuring ttys, A-38

W

Worksheet

local printer planning, C-2

remote printer planning, C-3

RS-232/422 port planning, B-6, C-5

terminal line planning, B-5, C-4

tty line planning, B-5, C-4

VAC/16 controller planning, B-7, C-5

VDA host adapter planning, B-9, C-6

Wrong characters printed, printer type, A-4

Wrong spacing, A-3

double spaced, A-4

letters run together, A-4

letters zigzag down page, A-4

no left margin, A-4

X

XOFF signal, 2-3

XON signal, 2-3

TIPS ORDERING PROCEDURES

TO ORDER

- 1 An order can be placed with the TIPS group in two ways:
 - a MAIL ORDER – Use the order form on the opposite page and fill in all requested information. Be sure to include shipping charges and local sales tax. If applicable, write in your tax exempt number in the space provided on the order form.
 - b Send your order form with payment to:
Data General Corporation
ATTN: Educational Services/TIPS G155
4400 Computer Drive
Westboro, MA 01581-9973
 - c TELEPHONE – Call TIPS at (508) 870-1600 for all orders that will be charged by credit card or paid for by purchase orders over \$50.00. Operators are available from 8:30 AM to 5:00 PM EST.

METHOD OF PAYMENT

- 2 As a customer, you have several payment options:
 - a Purchase Order – Minimum of \$50. If ordering by mail, a hard copy of the purchase order must accompany order.
 - b Check or Money Order – Make payable to Data General Corporation. Credit Card – A minimum order of \$20 is required for MasterCard or Visa orders.

SHIPPING

- 3 To determine the charge for UPS shipping and handling, check the total quantity of units in your order and refer to the following chart:

Total Quantity	Shipping & Handling Charge
1-4 Items	\$5.00
5-10 Items	\$8.00
11-40 Items	\$10.00
41-200 Items	\$30.00
Over 200 Items	\$100.00

If overnight or second day shipment is desired, this information should be indicated on the order form. A separate charge will be determined at time of shipment and added to your bill.

VOLUME DISCOUNTS

- 4 The TIPS discount schedule is based upon the total value of the order.

Order Amount	Discount
\$0-\$149.99	0%
\$150-\$499.99	10%
Over \$500	20%

TERMS AND CONDITIONS

- 5 Read the TIPS terms and conditions on the reverse side of the order form carefully. These must be adhered to at all times.

DELIVERY

- 6 Allow at least two weeks for delivery.

RETURNS

- 7 Items ordered through the TIPS catalog may not be returned for credit.
- 8 Order discrepancies must be reported within 15 days of shipment date. Contact your TIPS Administrator at (508) 870-1600 to notify the TIPS department of any problems.

INTERNATIONAL ORDERS

- 9 Customers outside of the United States must obtain documentation from their local Data General Subsidiary or Representative. Any TIPS orders received by Data General U.S. Headquarters will be forwarded to the appropriate DG Subsidiary or Representative for processing.

DATA GENERAL CORPORATION

TECHNICAL INFORMATION AND PUBLICATIONS SERVICE

TERMS AND CONDITIONS

Data General Corporation ("DGC") provides its Technical Information and Publications Service (TIPS) solely in accordance with the following terms and conditions and more specifically to the Customer signing the Educational Services TIPS Order Form. These terms and conditions apply to all orders, telephone, telex, or mail. By accepting these products the Customer accepts and agrees to be bound by these terms and conditions.

1. CUSTOMER CERTIFICATION

Customer hereby certifies that it is the owner or lessee of the DGC equipment and/or licensee/sub-licensee of the software which is the subject matter of the publication(s) ordered hereunder.

2. TAXES

Customer shall be responsible for all taxes, including taxes paid or payable by DGC for products or services supplied under this Agreement, exclusive of taxes based on DGC's net income, unless Customer provides written proof of exemption.

3. DATA AND PROPRIETARY RIGHTS

Portions of the publications and materials supplied under this Agreement are proprietary and will be so marked. Customer shall abide by such markings. DGC retains for itself exclusively all proprietary rights (including manufacturing rights) in and to all designs, engineering details and other data pertaining to the products described in such publication. Licensed software materials are provided pursuant to the terms and conditions of the Program License Agreement (PLA) between the Customer and DGC and such PLA is made a part of and incorporated into this Agreement by reference. A copyright notice on any data by itself does not constitute or evidence a publication or public disclosure.

4. LIMITED MEDIA WARRANTY

DGC warrants the CLI Macros media, provided by DGC to the Customer under this Agreement, against physical defects for a period of ninety (90) days from the date of shipment by DGC. DGC will replace defective media at no charge to you, provided it is returned postage prepaid to DGC within the ninety (90) day warranty period. This shall be your exclusive remedy and DGC's sole obligation and liability for defective media. This limited media warranty does not apply if the media has been damaged by accident, abuse or misuse.

5. DISCLAIMER OF WARRANTY

EXCEPT FOR THE LIMITED MEDIA WARRANTY NOTED ABOVE, DGC MAKES NO WARRANTIES, EXPRESS OR IMPLIED, INCLUDING, BUT NOT LIMITED TO, WARRANTIES OF MERCHANTABILITY AND FITNESS FOR PARTICULAR PURPOSE ON ANY OF THE PUBLICATIONS, CLI MACROS OR MATERIALS SUPPLIED HEREUNDER.

6. LIMITATION OF LIABILITY

A. CUSTOMER AGREES THAT DGC'S LIABILITY, IF ANY, FOR DAMAGES, INCLUDING BUT NOT LIMITED TO LIABILITY ARISING OUT OF CONTRACT, NEGLIGENCE, STRICT LIABILITY IN TORT OR WARRANTY SHALL NOT EXCEED THE CHARGES PAID BY CUSTOMER FOR THE PARTICULAR PUBLICATION OR CLI MACRO INVOLVED. THIS LIMITATION OF LIABILITY SHALL NOT APPLY TO CLAIMS FOR PERSONAL INJURY CAUSED SOLELY BY DGC'S NEGLIGENCE. OTHER THAN THE CHARGES REFERENCED HEREIN, IN NO EVENT SHALL DGC BE LIABLE FOR ANY INCIDENTAL, INDIRECT, SPECIAL OR CONSEQUENTIAL DAMAGES WHATSOEVER, INCLUDING BUT NOT LIMITED TO LOST PROFITS AND DAMAGES RESULTING FROM LOSS OF USE, OR LOST DATA, OR DELIVERY DELAYS, EVEN IF DGC HAS BEEN ADVISED, KNEW OR SHOULD HAVE KNOWN OF THE POSSIBILITY THEREOF; OR FOR ANY CLAIM BY ANY THIRD PARTY.

B. ANY ACTION AGAINST DGC MUST BE COMMENCED WITHIN ONE (1) YEAR AFTER THE CAUSE OF ACTION ACCRUES.

7. GENERAL

A valid contract binding upon DGC will come into being only at the time of DGC's acceptance of the referenced Educational Services Order Form. Such contract is governed by the laws of the Commonwealth of Massachusetts, excluding its conflict of law rules. Such contract is not assignable. These terms and conditions constitute the entire agreement between the parties with respect to the subject matter hereof and supersedes all prior oral or written communications, agreements and understandings. These terms and conditions shall prevail notwithstanding any different, conflicting or additional terms and conditions which may appear on any order submitted by Customer. DGC hereby rejects all such different, conflicting, or additional terms.

8. IMPORTANT NOTICE REGARDING AOS/VS INTERNALS SERIES (ORDER #1865 & #1875)

Customer understands that information and material presented in the AOS/VS Internals Series documents may be specific to a particular revision of the product. Consequently user programs or systems based on this information and material may be revision-locked and may not function properly with prior or future revisions of the product. Therefore, Data General makes no representations as to the utility of this information and material beyond the current revision level which is the subject of the manual. Any use thereof by you or your company is at your own risk. Data General disclaims any liability arising from any such use and I and my company (Customer) hold Data General completely harmless therefrom.

Installing and
Managing
Printers on the
DG/UX™ System

093-701132-01

Cut here and insert in binder spine pocket