

8.0 Release Notes

HP 9000 Series 600/800 Computers



HP Part No. B2437-90007

Printed in USA 1/91

HP Computer Museum
www.hpmuseum.net

For research and education purposes only.

Notices

The information contained in this document is subject to change without notice.

Hewlett-Packard makes no warranty of any kind with regard to this manual, including, but not limited to, the implied warranties of merchantability and fitness for a particular purpose. Hewlett-Packard shall not be liable for errors contained herein or direct, indirect, special, incidental or consequential damages in connection with the furnishing, performance, or use of this material.

Warranty. A copy of the specific warranty terms applicable to your Hewlett-Packard product and replacement parts can be obtained from your local Sales and Service Office.

Copyright © Hewlett-Packard Company 1991

This document contains information which is protected by copyright. All rights are reserved. Reproduction, adaptation, or translation without prior written permission is prohibited, except as allowed under the copyright laws.

Restricted Rights Legend. Use, duplication or disclosure by the U.S. Government Department of Defense is subject to restrictions as set forth in paragraph (b)(3)(ii) of the Rights in Technical Data and Software clause in FAR 52.227-7013.

Use of this manual and flexible disc(s) or tape cartridge(s) supplied for this pack is restricted to this product only. Additional copies of the programs can be made for security and back-up purposes only. Resale of the programs in their present form or with alterations, is expressly prohibited.

Copyright © AT&T, Inc. 1980, 1984, 1986

Copyright © The Regents of the University of California 1979, 1980, 1983, 1985

This software and documentation is based in part on the Fourth Berkeley Software Distribution under license from the Regents of the University of California.

Copyright © The Regents of the University of Colorado, a body corporate 1979

This document has been reproduced and modified with the permission of the
Regents of the University of Colorado, a body corporate.

Printing History

New editions of this manual will incorporate all material updated since the previous edition. The manual printing date and part number indicate its current edition. The printing date changes when a new edition is printed that contains significant changes from the previous edition. The manual part number changes when major technical changes occur between editions.

January 1991 ... Edition 1.

This is the only edition of this release notes file.

Contents

1. Introduction to the 8.0 Release Notes File	
Your Old Release Notes File Will Not be Overwritten	1-1
You Can Get a Printed Copy of this File	1-2
The Release Notes File is Not Your Only Source of Information .	1-2
How to Read and Use This Release Notes File	1-3
Finding the Information You Want	1-4
Release Notes Parts and Headings	1-5
Part 1: Products That Relate to Filesets	1-5
Part 2: Release-specific Topics	1-6
Part 3: Removed and Obsoleted Features	1-7
Begin Your Searches in Following Chapters	1-7
2. Part 1: Products That Relate to Filesets	
_AGP/DGL	2-1
_Allbase/SQL	2-2
_ANSI/C Compiler	2-2
OSF Data Alignment Pragmas	2-2
Instruction Scheduling	2-2
New Debug Information Format	2-3
\$ as Valid Identifier Character	2-3
Procedure Integration (optimization level 3)	2-3
_ARPA/9000	2-4
BIND (nameserver) changes	2-4
_Data Pair	2-5
_Fonts	2-6
_Fortran	2-7
Differences between 8.0 and 7.40 Fortran/300	2-7
II. SHARED LIBRARIES SUPPORT	2-7
III. POSIX-CONFORMING FORT77 COMMAND	2-7
IV. ALIGNMENT	2-7

V. STATIC ANALYSIS SUPPORT	2-8
VI. MESSAGE CATALOGS	2-8
VII. FCOPTS FEATURE	2-8
_GKS	2-9
Support	2-9
Performance	2-9
Dependencies	2-9
_HP-UX/License (kernel)	2-10
_HP Windows/9000	2-10
A) Support:	2-10
B) Performance	2-11
C) Defects	2-11
Obsolete X Windows Environment for S 300/400	2-11
Ignition Fileset in S 300/400 Product	2-11
Windows/9000 and Starbase	2-12
_LAN/9000	2-13
_NCSNCK-RUN Release 1.5.1	2-21
_Network Services	2-32
_NFS/9000	2-32
NFS/9000 HP-UX File System	2-32
NFS/9000 Performance Features and Tuning	2-32
Performance Improvements	2-32
NIS-maps	2-33
NFS/9000 HP-UX Daemons	2-33
_NLIO	2-34
Content	2-34
Dependencies	2-34
Impact	2-34
New Features	2-35
Deleted Features	2-35
_NLS	2-36
NLS Enhancement: EUC/UJIS Support	2-36
NLS Enhancement: ISO 8859-1 support	2-36
NLS Enhancement: ISO 8859-2 ISO 8859-5 support	2-36
NLS Tools	2-37
Buildlang(1M)	2-37
Locale.def file	2-37
All currently loaded language filesets must be updated.	2-37

JAPANESE fileset	2-38
Nl_tools_16(3C)	2-38
_Pascal	2-39
What's new for 8.0?	2-39
Command line options	2-39
Directives	2-39
What's changed between 7.40 and 8.0?	2-39
_SBDL	2-40
_Software Dvmt Toolkit	2-40
What	2-40
Why	2-40
What you need to do	2-40
_Starbase	2-41
Starbase Support for HP98705 Display	2-43
2. Drivers	2-43
Starbase Support for HP98736 Display	2-44
Texture Mapping	2-46
With data primitives	2-46
Contouring	2-47
Deformation	2-47
Anti-aliasing	2-47
Alpha transparency	2-47
12 bit indexing	2-47
2. Drivers	2-48
Starbase Support for the 98705A/B/C Graphics Subsystem	2-48
Alternatives	2-52
Starbase Support for HPA1096 Display	2-52
1. Supplementary Programming Information	2-52
Starbase Support for HPA1416 (CH')	2-53
Compatibility	2-53
Workarounds	2-55
Using FA/FM with the SOX11 Driver	2-55
Starbase Display List	2-55
Reconfigurable Operating Parameters	2-56
Enhanced Memory Manager	2-56
Correction to Rename Segment (and References)	2-56
Shared Libraries	2-57
Compile/Link Sequence	2-57

GOPEN	2-57
inquire_device_driver	2-57
inquire_capabilities	2-57
Starbase and ANSI C	2-58
Starbase Last Support for Some Software	2-58
_Switchover/UX	2-59
Summary of Change	2-59
Configuration	2-59
Features	2-59
Performance data	2-59
Size requirement	2-59
_X.25/9000	2-60
New X.25 Installation Procedure	2-60
Changes for X.25 Programmatic Access	2-60
Changes for Network Logging and Tracing for X.25	2-61
Changes for x29server	2-62
New Features in X.25 Programmatic Access	2-63
New Features in IP-over-X.25	2-64
New X.25 Installation Procedure	2-64
New X.25 PAD products for 8.0	2-64
New Command: "x25lbttest" on S300/S400 only	2-64
_X11 Windows Developer's Toolkit	2-65
Change in Motif and X11 include files and library locations	2-65
What	2-65
Why	2-65
What you need to do	2-65
_X11 Window System	2-66
Visual User Environment (HP VUE)	2-66
HP_VUE_Login_Manager	2-66
HP_VUE_Session_Manager	2-67
HP_VUE_Workspace_Manager	2-67
HP_VUE_Help_Manager	2-68
HP_VUE_Style_Manager	2-68
HP_VUE_Utility_Clients	2-68
HP_VUE_API_Library_(libXv.a)	2-69
OSF/Motif 1.1	2-69
OSF/Motif_1.1_ToolKit_Content	2-69
Style_Guide_Conformance	2-70

CascadeButton	2-70
List Supports	2-70
Text Supports	2-71
ScrolledWindow	2-71
Bulletin_Board	2-71
Buttons	2-71
Main_Window	2-71
Paned_Window	2-71
ScrollBar	2-71
Scale	2-71
Compound_string_and_Fontlist_Performance	2-72
Resolution_Independence	2-72
Toolkit New_features	2-72
New_Convenience_functions	2-72
New Resources and Functions	2-73
Label_and_Buttons	2-73
Text Implement QueryGeometry	2-73
Shells	2-73
Manager	2-73
ScrolledWindow	2-73
OptionMenu	2-74
BulletinBoard	2-74
Form Implements QueryGeometry.	2-74
FileSelectionBox	2-74
Primitive	2-74
Gadget	2-75
Scrollbar	2-75
Scale	2-75
XmNprocessingDirection	2-75
New Widgets for Performance	2-75
Window_Manager_Contents	2-75
X11 Server for the 8.0 Release	2-76
8.0 X11 Server Changes and Features	2-76
Backward Compatibility Mode	2-77
SaveUnders	2-78
Scalable Typeface Support	2-78
xwcreate and gwind Changes	2-79
Non-Rectangular Window (Shape) Extension Support	2-80

Standard Input Extension Support	2-80
Incorrect rgb.txt Entries	2-80
Shared Memory Transport Support for the X Server	2-80
Using the Shared Memory Transport Facility	2-81
Advantages of the Shared Memory Transport Facility	2-81
Disadvantages of the Shared Memory Transport Facility	2-81
X11 Window Documentation	2-82
PAM (Personal Applications Manager)	2-83
3. Part 2: Release-specific Topics	
_Backup and Recovery of Files	3-1
fbbackup(1M) and frecover(1M) changed to support MO and DAT media	3-1
Support for MO devices	3-1
Support for DAT devices	3-1
fbbackup(1M) and frecover(1M) specific changes	3-2
fbbackup(1M) and frecover(1M) compatibility	3-2
_Computer Languages	3-3
ANSI/C Compiler	3-3
OSF Data Alignment Pragmas	3-3
Instruction Scheduling	3-3
New Debug Information Format	3-3
\$ as Valid Identifier Character	3-3
Procedure Integration (optimization level 3)	3-4
Support for Shared Libraries	3-4
Fortran	3-4
Differences between 8.0 and 7.40 Fortran/300	3-4
Pascal	3-5
Symbolic Debugger, Series 300/400	3-6
Shared libraries	3-6
ANSI/C and Debugger	3-6
C++ and Debugger	3-7
Other Enhancements	3-8
Compatibility	3-8
Impact on Performance	3-9
User Limitations	3-9
Symbolic Debugger, Series 600/800	3-9
Shared libraries	3-9

C++ and Debugger	3-10
PA-RISC Float Register display	3-11
Other Enhancements	3-11
Compatibility	3-12
Impact on Performance	3-12
Support	3-13
User Limitations	3-13
Documentation	3-14
HP 9000 Computers Series 600/800 Documentation	3-15
System Documents for Series 600/800	3-15
Graphics Documents for Series 600/800	3-19
Other HP-UX Learning Products for Series 600/800	3-20
HP 9000 Computers Series 300/400 Documentation	3-21
System Documents	3-21
Graphics Documents for Series 300/400	3-25
Language Documents for Series 300/400	3-27
Other Learning Products for Series 300/400	3-28
Networking Documents for HP 9000 Systems	3-28
Drivers	3-30
Autochanger driver (C1700A Optical)	3-30
Summary of Change	3-30
Configuration	3-30
Features	3-30
Impact	3-31
Performance data	3-31
Size requirement	3-32
centronics (HP-PB)	3-32
Summary of Change	3-32
Configuration	3-32
Features	3-32
Impact	3-32
CS80 device driver (disc1)	3-33
Summary of Change	3-33
Configuration	3-33
Features	3-33
Impact	3-33
Size requirement	3-33
SCSI Supported on Series 800	3-34

Content	3-34
Compatibility	3-34
Dependencies	3-35
User Limitations	3-35
_File Systems and Files	3-36
fs_async and Synchronous Disk Writes	3-36
Features	3-36
Configuration	3-36
Impact	3-36
Performance data	3-37
Disk Quotas and Commands	3-37
Added Functionality	3-37
HP Enhancements	3-37
Implications with respect to other features	3-38
Performance Impact as a result of Disk Quotas	3-38
General Overhead	3-38
Configuration on a file system	3-39
Effect of quotas for individual users	3-39
System Boot Overhead	3-39
Additional Run-time Overhead	3-39
Disk Space Requirements	3-40
/quotas file	3-40
/etc/mnttab, /etc/checklist, getmntent(3)	3-40
_HP OSF/Motif	3-41
_HP-UX Clusters	3-41
A Cluster and Kernel Memory	3-41
Standalone System	3-41
Diskless client with small amount of memory	3-42
Client with Locally Mounted File System	3-42
Server	3-43
Locally Mounted File Systems	3-43
Summary of Change	3-43
Machines Affected	3-43
Features	3-43
LMFS and NFS	3-44
Global visibility	3-44
/etc/mnttab	3-44
Shutdown(1m) and reboot(1m)	3-44

Impact	3-45
Crash recovery	3-47
Open files	3-47
FIFO's	3-47
Current working directories	3-47
Sockets	3-48
File Locks	3-48
NFS mounts	3-48
Performance data	3-48
Size requirements	3-48
Support for auxiliary servers in HP-UX clusters	3-48
Summary of Change	3-48
Information	3-49
Configuration	3-49
Features	3-49
Impact	3-50
_HP-UX Commands	3-51
New Commands for 8.0	3-51
The AWK Command	3-51
Summary of Change	3-51
Lvalues referencing awk fields	3-51
New reserved words	3-52
Specifying input field separator (FS)	3-53
Extended regular expressions	3-53
Non-existent fields	3-53
Configuration	3-53
Impact	3-54
Alternatives	3-54
The CAT Command	3-54
Summary of Change	3-54
The CHGRP CHMOD CHOWN Commands	3-54
Summary of Change	3-54
Alternatives	3-54
Performance data	3-54
The COMM Command	3-54
Summary of Change	3-54
The DD Command	3-55
Alternatives	3-55

The ED Command	3-55
Summary of Change	3-55
Impact	3-55
The ELM Command	3-55
The FIND Command	3-56
Summary of Change	3-56
Features	3-56
The FSCK Command	3-56
Summary of Change	3-56
Configuration	3-56
Features	3-57
Impact	3-57
Alternatives	3-57
The GETCONF Command	3-57
Summary of Change	3-57
Features	3-57
Impact	3-57
The GREP EGREP FGREP Commands	3-57
Summary of Change	3-57
Features	3-58
Impact	3-58
The ID Command	3-58
Summary of Change	3-58
Impact	3-58
Alternatives	3-58
The KSH Command	3-58
Summary of Change	3-58
Features	3-59
Support for multiple co-processes	3-59
Compatibility Issues	3-60
The LN Command	3-60
Summary of Change	3-60
Features	3-60
Impact	3-60
The LP Command (Spooler)	3-60
Summary of Change	3-60
Impact	3-61
The LS Command	3-61

Summary of Change	3-61
Impact	3-61
The MAKE Command	3-61
Summary of First Change	3-61
Features	3-61
Impact	3-61
Alternatives	3-62
Summary of Second Change	3-62
Impact	3-62
Alternatives	3-62
The MKBOOT Command	3-63
The MONITOR Command	3-63
A New Command	3-63
The RM Command	3-63
Summary of Change	3-63
The RMDIR Command	3-64
Summary of Change	3-64
The SORT Command	3-64
Summary of Change	3-64
Impact	3-64
The TOP Command	3-64
A New Command	3-64
The UNIQ Command	3-64
Summary of Change	3-64
Impact	3-65
The UUENCODE Command	3-65
Summary of Change	3-65
Impact	3-65
The UUDECODE Command	3-65
Summary of Change	3-65
Impact	3-65
_HP-UX Kernel	3-66
Autoconfiguration/800	3-66
Summary of Change	3-66
Configuration	3-66
Features	3-66
Non-Features	3-66
Impact within the kernel	3-67

Impact outside the kernel	3-67
Alternatives	3-68
Greater Than 60 Open Files	3-68
Summary of Change	3-68
Configuration	3-69
Features	3-69
Impact	3-69
Alternatives	3-69
Header Files	3-71
<curses.h>	3-71
Libraries Used by the Kernel (e.g. "srvtol()", "closedir()") . .	3-72
Shared Library Utilization (Core Commands and Libraries)	3-72
Summary of Change	3-72
Configurations Supported	3-72
Lockable Memory	3-72
Availability on an 8 MB System	3-72
Possible Problem with Memory Locking	3-73
Shared Libraries	3-74
Summary of Change	3-74
Configuration	3-74
Features	3-74
Impact	3-74
Size Requirement	3-74
System V IPC	3-74
Summary of Change	3-74
Features	3-75
Impact	3-75
Swapping Subsystem	3-75
Summary of Change	3-75
Configuration	3-75
Virtual Memory	3-76
Summary of Change	3-76
Configuration	3-76
Features	3-76
Impact	3-77
_Installing HP-UX	3-79
S800 Install	3-79
Summary of Change	3-79

Impact	3-79
808/815 HP-UX Release 8.0 Install/Update Notes	3-79
Instant Ignition	3-81
Obsolete X Windows Environment for S 300/400	3-81
Ignition Fileset in S 300/400 Product	3-81
LP Spooler System	3-82
centronics (HP-PB)	3-82
Summary of Change	3-82
Configuration	3-82
Features	3-82
Impact	3-82
Spooling on cnodes	3-82
Online Diagnostics	3-83
Series 300 Computers	3-83
The Series 300 has Online Diagnostics	3-83
System Configuration and Dependencies	3-83
System Configuration and Dependencies	3-83
System Configuration and Dependencies	3-84
Enhancements and New Developments	3-84
Files Added/Deleted	3-84
Special Installation/Execution Instructions	3-85
Enhancements and New Developments	3-85
Known Problems	3-85
PAM (Personal Applications Manager)	3-88
Performance and Tuning	3-88
fs_async and synchronous Disk Writes	3-88
Features	3-88
Configuration	3-88
Impact	3-89
Performance data	3-89
Powerfail Note and SCSI	3-89
SCSI	3-90
C1700A Optical Autochanger driver	3-90
Summary of Change	3-90
Configuration	3-90
Features	3-90
Impact	3-91
Performance data	3-91

Size requirement	3-92
User Limitations for S800 SCSI	3-92
_Standards and HP Conformance	3-93
_STD/FES	3-94
NOTE 1	3-94
NOTE 2	3-94
_Swap Space	3-95
Swapping Subsystem	3-95
Summary of Change	3-95
Configuration	3-95
_System Administration Manager (SAM)	3-96
New X.25 Installation Procedure	3-96
Configuration of X.25/9000 through SAM utility	3-96
SAM Backup and Recovery	3-96
Cluster Configuration with Domain Names and Name Server	3-97
SAM and Support for Designating Swap Locations	3-97
Miscellaneous Bug Fixes in SAM	3-97
SAM Kernel Configuration	3-98
New Hardware for SAM	3-98
SAM and VM System Support	3-99
SAM and Series 800 Auto-Config Support	3-99
SAM and Peripheral Devices	3-99
SAM and SCSI Disk/CD-ROM Support	3-100
SAM and Non-HP Terminal Support	3-100
SAM and Network Configuration	3-100
SAM and Other Utilities	3-101
SAM and User Account UID Specification	3-101
SAM and File System Protection When Removing Users	3-102
SAM and Task Customization	3-102
You Cannot Use SAM Programmatically	3-103
SAM Responds at an Interactive Speed	3-103
_System Verifier	3-104
_Terminal I/O	3-104
New Features	3-104
Autoconfiguration	3-104
CIO 16-channel MUX	3-104
NIO 8-channel AP MUX	3-104
DIO 8-channel MUX	3-105

Compatibility	3-105
Autoconfiguration	3-105
CIO 16-channel MUX	3-105
NIO 8-channel AP MUX	3-105
DIO 8-channel MUX	3-106
Defect Fixes	3-106
Improved Modem Processing on CIO Machines	3-106
Obsolescence	3-106
_Updating HP-UX	3-107
808/815 HP-UX Release 8.0 Install/Update Notes	3-107

4. **Part 3: Removed or Obsolete Products or Features**

_AGP/DGL	4-1
_HP Windows/9000	4-1
_Powerfail Note and SCSI	4-2
_RFA	4-2
_X.25/9000	4-2

Introduction to the 8.0 Release Notes File

This release notes file contains information you may want to examine after you update or install the 8.0 release of HP-UX. The file contains the following types of information:

- New features in the 8.0 release. This includes new software that was added to HP-UX and new functionality that was added to an existing part of HP-UX.
- Changed features in the 8.0 release. This information ranges in level of detail from low-level descriptions of changes in a system call to high-level descriptions of changes in the features of a product.
- Removed or deleted features. This information consists of simple explanations of any part of HP-UX that has been removed. It also includes a mention of features that will disappear in future releases.

Your Old Release Notes File Will Not be Overwritten

If you updated HP-UX from a previous release to the 8.0 release, the update process should not have overwritten the existing release notes file. You should be able to make any comparisons you wish between this file and the previous release notes file.

You Can Get a Printed Copy of this File

HP produces a “hard-copy” version of this file and, hopefully, you had an opportunity to review that document before you updated or installed the 8.0 release. Since the “hard-copy” was produced slightly ahead of this file, the possibility exists that the contents may not be identical.

The Release Notes File is Not Your Only Source of Information

Be aware that you should have other sources of information about the 8.0 release. That information supplements the information in this file.

README	These documents come with the products you purchased. They contain information about the install/update processes.
Documentation for HP-UX	This set of documentation contains the wealth of information you can use to get maximum benefit from using HP-UX. Use these documents as necessary after you work through the above documents. The “Finding Information” document, which is part of the set, describes each document in the set.
Software Status Bulletin	This document contains information about currently outstanding problems and enhancement . You can get information, for example, about a known problem related to an HP-UX subsystem.

How to Read and Use This Release Notes File

The **8.0ReleaseNotes** file contains information about the parts of the HP-UX operating system (kernel, commands, Starbase, X11, LP Spooler, and so on). The release notes are contained in parts:

- | | |
|--------|--|
| Part 1 | Fileset related products (for example, NLIO, X11, and AGP/DGL). The information relates directly to the products you purchase. Look at Chapter 2. |
| Part 2 | Information about 8.0 parts or constructs that may cut across products or simply be separated from them in terms of presenting information (for example, CD-ROM, Kernel, and Swap space). Look at Chapter 3. |
| Part 3 | Removed or obsoleted products or features (for example, RFA). This information may also be mentioned in other parts, but this part helps you focus on removed functionality. Look at Chapter 4. |

Finding the Information You Want

Each topic in a part is a heading. The topic also appears between dashed lines and is preceded by an underscore.

The following items explain how to find topics and get information:

1. Information appears under topics (headings). A list (shown below) has the topics. You may want to note topics for which you want information because your screen will probably not show all of them.
2. Position the cursor below the list and search on a topic for which you want information. For example, if you issued the *more*(1) command to do a search for information about the "Starbase" product, you could type:

/_Starbase Return

The search will take you to the beginning of the information, which you can read according to the operations performed by *more*(1).

3. Read the information that appears under the topic as desired, making notes according to your needs.
4. Repeat steps 2 and 3 to get information according to the topics in your list, being aware that you have other sources of information (for example, the documentation set for HP-UX). You may need to relate information contained in this file to information you get from other sources.

Release Notes Parts and Headings

The 8.0ReleaseNotes file has three parts, and each part has topics.

Part 1: Products That Relate to Filesets

Part 1 has information about the products you purchased. Use the topic name, with the underscore, as the string a search.

- _AGP/DGL
- _Allbase/SQL
- _ANSI/C Compiler
- _ARPA/9000
- _Data Pair
- _Fonts
- _Fortran
- _GKS
- _HP-UX/License (kernel)
- _HP Windows/9000
- _LAN/9000
- _NCSNCK-RUN
- _Network Services
- _NFS/9000
- _NLIO
- _NLS
- _Pascal
- _SBDL
- _Software Dvmt Toolkit
- _Starbase
- _Switchover/UX
- _X.25/9000
- _X11 Windows Dev. Toolkit
- ._X11 Window System

Part 2: Release-specific Topics

Part 2 has information related to HP-UX entities used by the system. Use a topic names, with the underscore, as the string for a search.

- _Backup and Recovery of Files
- _Computer Languages
- _Documentation
- _Drivers
- _File System and files
- _HP OSF/Motif
- _HP Support
- _HP-UX Clusters
- _HP-UX Commands
- _HP-UX Daemons
- _HP-UX File System
- _HP-UX Kernel
- _Installing HP-UX
- _Instant Ignition
- _LP Spooler System
- _Online Diagnostics
- _PAM (Per Applic Mgr)
- _Performance Features/Tuning
- _Powerfail Note and SCSI
- _SCSI
- _Standards and HP Conformance
- _STD/FES
- _Swap Space
- _System Administration Manager (SAM)
- _System Verifier
- _Terminal I/O
- _Updating HP-UX

Part 3: Removed and Obsoleted Features

Part 3 (below) has information related to entities that are not supported in 8.0 or will not be supported beyond 8.0. Use the names as the string you specify for a search. Including the underscore (_) and the rest of the name will take you to the beginning of the information.

```
_AGP/DGL
_HP Windows/9000
_Powerfail Note and SCSI
_RFA
_X.25/9000
```

Begin Your Searches in Following Chapters

The following chapters contain information according to the above topics.

- Chapter 2 contains information for Part 1.
- Chapter 3 contains information for Part 2.
- Chapter 4 contains information for Part 3.

Part 1: Products That Relate to Filesets

This part has information about products a customer may purchase.

_AGP/DGL

A) Support:

- Support for AGP/DGL on the S800 has been increased for 8.0 to include the following terminal and plotter drivers:

Terminals:	A0001, K0001
HP7550A:	D0047, A0047, B0047, L0046
HP7580A,B/85A,B:	D0006, L0006, P0006
HP7580B/85B/86B:	D0048, L0006

- AGP/DGL is being discontinued after 8.0; that is, 8.0 will be the last supported release for AGP/DGL.

All customers will be given a last chance to purchase this product if they have applications that they wish to port to another API but have not yet had the opportunity to do the port.

Be assured that HP will continue to protect customer investment by offering support of AGP/DGL on HP-UX 8.0 for 5 years.

B) Performance

- AGP/DGL 8.0 performance is at least that of the HP-UX 7.0 release on the S800 workstations.

C) Defects

- At this time, there are no known defects associated with AGP/DGL.

_Allbase/SQL

“Summary of Change”

New features are being added but none are expected to impact existing applications. No changes are planned which, by themselves, should cause an existing application to have to recompile or relink. However, changes elsewhere in the system may create this need.

_ANSI/C Compiler

For the Series 800, the following change was made.

User visible options for in-line compiler directive/pragma:

```
#pragma HP_DESTINATION ARCHITECTURE [1.0][1.1]
#pragma HP_DESTINATION SCHEDULER [1.0][1.1]
```

For the Series 300, the following changes were made.

OSF Data Alignment Pragmas

The 8.0 Series 300 C compiler provides a new pragma (HP_ALIGN) to align structures to be compatible with the Series 500, Series 800, DOMAIN series 3000/4000, DOMAIN series 10000, and provide an alignment mode intended to be the default for OSF.

Instruction Scheduling

For the series 300 8.0 release c2 (the peephole optimizer) now performs instruction scheduling. The scheduler takes advantage of the MC68000's ability to concurrently execute floating point and integer instructions, and attempts to avoid sequences that cause A-register or FP-register interlocks which result in a pipeline stall.

New Debug Information Format

In the 8.0 release, the series 300 C compiler generates a new format for debugging information which is consistent with that of the series 800. This new format is inconsistent with prior Series 300 releases. Consequently, programs compiled with debug information must be recompiled to work correctly with the 8.0 debuggers.

\$ as Valid Identifier Character

To ease portability of C source from DOMAIN to HP-UX platforms, the 8.0 Series 300 C compiler now accepts “\$” as a valid identifier character (Note, not as the first character of an identifier). “\$”s are unconditionally accepted in compatibility mode and conditionally accepted in ansi mode (the “+e” cc option enables its acceptance in ansi mode).

Procedure Integration (optimization level 3)

Available only with the C/ANSI C Product.

The S300 8.0 C compiler supports a new optimization level “+O3” which includes all of the level 2 optimizations, and in addition performs procedure integration. Procedure integration will replace certain function calls with the actual function code, which can increase performance by removing the call overhead and allowing more optimizations on the in-lined code. Several pragmas (HP_INLINE_LINES, HP_INLINE_FORCE, HP_INLINE_OMIT, HP_INLINE_DEFAULT, and HP_INLINE_NOCODE) have been added to provide finer control on procedure integration.

_ARPA/9000

There are three new ARPA services in HP-UX 8.0.

BootP (Bootstrap Protocol) allows HP's 700/X Terminal to discover its IP address, subnet mask, name server, and boot file. Bootp is implemented in the server `/etc/bootpd` (see `bootpd(1M)`) which is based on CMU's version 2.1 BootP server. The tool `/etc/bootpquery` (see `bootpquery(1M)`) allows you to simulate BootP clients to diagnose server configurations.

TFTP (Trivial File Transfer Protocol) allows the HP 700/X Terminal to load fonts and other boot information. TFTP is implemented in a server, `/etc/tftpd`, and client, `/usr/bin/tftp` (see `tftpd(1M)` and `tftp(1)`). Both of these are based on the BSD 4.3 implementations.

The `finger(1)` command now allows queries over the network. The server `/etc/fingerd` (see `fingerd(1M)`) allows remote users to access finger information on your system.

The 8.0 ARPA services are based on recent revisions of BSD 4.3 ARPA services. See "Using ARPA Services", "Installing and Administering ARPA Services", and individual man pages for details of new or different functionality.

BIND (nameserver) changes

BIND (nameserver): TXT records are handled differently by the nameserver in 8.0. TXT records will not be properly exchanged between a 7.0 name server and an 8.0 name server. The reason for this change is that the 7.0 name server was not following the specifications according to the RFCs. The nameserver in 8.0 is based on the 4.8.3 release of BSD's nameserver.

_Data Pair

The following items are new for 8.0.

- The maximum number of mirrors increased from 128 to 900.
- We have a new hpuxboot flag, “-mx”, which boots off whatever root mirror half is accessible. This can cause data loss.
- We allow a first-time mirrored root to boot even if both sides are not accessible. The consequences of the online half becoming inaccessible and the inaccessible half becoming accessible and booting are left to the user.
- The syntax of describing the root and swap mirrors in the “uxgen” file have changed slightly due to autoconfig.
- The device number of “/dev/rdisk/mirconfig” has changed.
- Errors to the ONLINE side of a reimaging mirror will cause the reimage to fail without the fail flag being set. Errors to the REIMAGE side set the fail flag as before.

_Fonts

The X Window System now supports scalable typefaces through the existing X interface. The scalable typeface support provided is through the Intellifont* technology from Agfa Compugraphic. This is the same scalable typeface technology used in PCL 5 printers. The scalable outlines bundled with HP-UX include “cg times”**, “univers”**, and “courier”. These typefaces can be accessed through the standard XLFD names. For example you can view a 15 point Univers typeface by typing:

```
xfd -fn "-cg-univers-***-**-150-***-***-***"
```

You can query the X Window System for XLFD strings that describe available scalable outlines by typing:

```
xlsfonts | grep 0-0-0-0
```

Note that this also allows you to purchase scalable typefaces for the LaserJet family of printers and to use those scalable outlines in the X Window System. See the documentation for “stmkfont”, “stmkdirs”, “stload”, “stlicense”, and “stconv” for further details.

The Intellifont Scalable Typeface Library from Agfa Compugraphic includes more than 180 different designs. Forty-six individual volumes of 4 faces are available as well as six application-specific collections for all office publishing needs. Please call Agfa Compugraphic directly at 1-800-873-FONT (3668) for a free brochure or contact your local Hewlett-Packard Sales and/or Customer Service office for more information.

- “Intellifont” is a registered trademark of Agfa Corporation.
- “CG Times”, based on Times New Roman under license from the Monotype Corporation plc, is a product of Agfa Corporation.
- “Univers” is a registered trademark of Linotype AG and/or its subsidiaries.

_Fortran

Differences between 8.0 and 7.40 Fortran/300

I. NEW I/O LIBRARY

A new I/O library, compatible with the Series 700/800, is provided with this release. This library is the default I/O library. In addition to the compatibility across platforms, the key benefit of this library is that recursive I/O is permitted. For example, a WRITE statement can contain a function call that itself performs a WRITE.

The library that previously was the default is provided for compatibility. To use the compatibility library, use the “+I300” option. The two I/O libraries are not compatible. Mixing calls to both libraries in the same program is likely to cause unpredictable results.

II. SHARED LIBRARIES SUPPORT

FORTTRAN/300 supports shared libraries. Two options are provided for creating shared libraries from Fortran source files. A shared library version of each of the Fortran libraries is provided to support library calls (e.g. I/O, Intrinsics) from user-created shared libraries.

III. POSIX-CONFORMING FORT77 COMMAND

The command “fort77” will provide a POSIX-conforming interface to the FORTRAN/300 compiler. When fort77 is used, two command-line options behave differently (“-L”, “-N”). One additional option is available when “fort77” is used (“-V”).

IV. ALIGNMENT

By default, “REAL*8” and “COMPLEX*16” items outside of structures on the stack or in saved data are aligned on 8-byte boundaries. The “+A3” and “+A8” options cause “REAL*8” and “COMPLEX*16” outside of structures to be aligned on 8-byte boundaries. These changes should not be visible to a correctly written program.

V. STATIC ANALYSIS SUPPORT

The “-y” option provides support for Softbench static analysis. This feature was released with 7.40, but will first be available with SoftBench 8.0.

VI. MESSAGE CATALOGS

The compile-time messages issued by the driver and front end are in message catalogs.

VII. FCOPTS FEATURE

The “|” character in the FCOPTS environment variable causes options that appear after the “|” to be recognized after the command line options.

_GKS

Support

HPGKS now includes full support for shared libraries. The GKS shared library is contained within a second fileset, called HPGKS-SHLIBS, and consists of the library called “libgks.sl”.

This new library replaces the “libgks1.a” and “libgks2.a” archive libraries in the typical GKS link sequence. The Starbase shared library, “libsb.sl”, replaces the “libsb1.a” and “libsb2.a” archive libraries in the link sequence.

See the HPGKS Users Guide for a more detailed description on using shared libraries and for examples on compiling and linking shared executable programs.

Performance

Shared libraries bind at run time. Hence, there is some additional overhead when compared to running an executable which had all of its archive libraries bound at link time. However, the penalty is small and shared libraries have been shown to provide 90% of the performance of the historical Starbase archive library executables.

Dependencies

HPGKS release 8.0 must be used in conjunction with Starbase release 8.0.

_HP-UX/License (kernel)

See “_HP-UX Kernel” in Part 2.

_HP Windows/9000

A) Support:

- Support for HP Windows/9000 specifically does not include the S400 machines.
- HP Windows/9000 has been unbundled from HP-UX for the 8.0 release. Customers wishing to receive the 8.0 version of this product must purchase it separately. For customers with support contracts, the new release may be obtained through their Support Engineer.
- When loading this product onto a system, requesting all filesets will cause update to load the following:

WIN-RUN
WIN-PRG
WIN-DEMO
NJWSERV

The NJWSERV fileset is for users that wish to use the Kanji fonts and should only be loaded if those fonts will be used.

To remove the NJWSERV fileset, as root, use the rmfn command and then re-execute the WIN-RUN customize script to ensure a consistent system.

- HP Windows/9000 is being discontinued after 8.0; that is, 8.0 will be the last supported release for HP Windows/9000.

All customers will be given a last change to purchase the product if they have applications that they wish to port to X11 but have not yet had the opportunity to do the port.

Be assured that we will continue to protect customer investment by offering support of HP Windows/9000 on HP-UX 8.0 for 5 years.

2-10 Part 1: Products That Relate to Filesets

B) Performance

- HP Windows/9000 8.0 performance is at least that of the HP-UX 7.0 release on the S300 workstations.

C) Defects

- At this time, there are no known defects associated with HP Windows/9000

Obsolete X Windows Environment for S 300/400

The X Windows Environment (fileset XENVIRON or XENV_CORE) based on HP-UX 7.0 and available for the Series 300/400 is no longer supported on HP-UX 8.0. Users of this environment are encouraged to switch to the Visual User Environment, a part of the X11-RUN fileset, and the IGNITION fileset. When the IGNITION fileset is loaded, the obsolete files in XENVIRON and XENV_CORE are removed. Refer to the manual “Installing and Updating HP-UX” for the Series 300/400 or the file `/etc/newconfig/Ignition/README`.

Ignition Fileset in S 300/400 Product

The IGNITION fileset has been added to the Runtime Product for the Series 300/400 (68K-based) Runtime Product. This fileset provides tools that configure the system to launch the Visual User Environment (VUE) when the system starts up. Usability tools such as `datebook`, `xcalendar`, `xdialog`, and `xhpcalc` are included in the IGNITION fileset. Refer to the manual “Installing and Updating HP-UX” for the Series 300/400 or the file `/etc/newconfig/Ignition/README` for more information on the IGNITION fileset.

Windows/9000 and Starbase

WINDOWS/9000 is adding the NJWSERV fileset which contains the kanji input support. This will result in an estimated 1Mbyte size increase for those loading the entire windows product. If not required the NJWSERV fileset should not be loaded or may be sysrm'd following the installation.

With the NJWSERV fileset, HP WINDOWS/9000 will require approximately 3 Mbytes. Without NJWSERV it requires 2 Mbytes.

The Kanji text input windows product previously provided by UXL on the NLIO product tape is now part of the Windows/9000 product. This is contained in the NJWSERV fileset.

The new drivers will not be supported by Windows/9000.

Windows/9000 programs will not run on the new HP98705 or HP98736 displays.

The X-window system should be used for all new applications and old applications should be ported.

Installing 8.0 HP-UX will not update the Windows/9000 system.

Windows/9000 will be available as a distinct product which must be purchased separately from the HP-UX update.

_LAN/9000

There have been several main changes in the LAN/Link product for 8.0.

I. One major 8.0 change is that the networking portion of the kernel is now based on 4.3 BSD Tahoe; including algorithms for improving networking performance. These algorithms include:

- slowstart
- congestion avoidance
- RTT estimation
- exponential retransmit backoff
- retransmit interval clamping
- dynamic window sizing
- aggressive ACK policy
- header prediction
- Internet Hosts Requirements Document segment sizing



In conforming with the 4.3 BSD standard, changes, both additions and deletions, were made to existing commands and options. The following is a list of the changes.

- In 8.0, three new BSD systems calls are supported: “sendmsg()”, “recvmsg()”, and “socketpair()”. Refer to the appropriate man pages for detailed descriptions of the use of these system calls.
- “select()” on BSD sockets: In 8.0, “select()” for read or write will also return true if an error exists on the socket. Also, “select()” for exceptional conditions returns true if out-of-band data is available for reading on the socket.
- “setsockopt()” system call argument list: In 7.0 “setsockopt” does not require a value for “optval” field for some of the options (like SO_REUSEADDR) and you could only set this option. Now in 8.0, the size of “optval” should be at least the size of an integer. Therefore you can both set and unset the SO_REUSEADDR option for the socket by toggling.

In 8.0, The `SO_LINGER` option is a combination. It sets a linger value, and also toggles linger behavior on and off. For `SO_LINGER`, “optval” points to a struct `linger`, defined in “/usr/include/sys/socket.h”. The `linger` structure contains an integer boolean flag to toggle behavior on/off, and an integer linger value.

- “setsockopt()” options removed: In 8.0, the options `SO_BURST_IN` and `SO_BURST_OUT` are irrelevant and are supported only in backward compatibility mode. If encountered, these options are ignored and no error is returned.

The `SO_DONTLINGER` option which previously turned off the `SO_LINGER` option has been removed from 8.0. In 8.0, this option is turned off by calling “setsockopt()” using `SO_LINGER` with the linger disabled.

- Unix Domain socket path name: (“sun_path” field in “struct” “sockaddr_un”): The path name of the Unix Domain socket was 108 bytes in length in pre-8.0 systems. In 8.0, it is limited to 92 bytes in length.
- “ioctl(SIOCSPGRP)” semantics changed: The semantics of the call to enable a socket to receive `SIGURG` using “ioctl(SIOCSPGRP)” have changed. Previously, if a negative process ID was passed to the “ioctl()”, it meant that the signal should be sent only to the process identified by that process ID. A positive process ID was interpreted as a process-group-ID and that the signal should be sent to the entire group with that process-group-ID. In 8.0, the semantics are reversed: a negative process ID is interpreted as a process-group-ID and a positive process ID is interpreted as a single process ID.

II. The various LAN features that were formerly configured as a unit may now be flexibly configured. This feature allows the user to save disc space and memory usage by installing and/or generating a networking kernel that includes just the desired LAN features.

If the user chooses to install selected LAN software components, then the following table lists the supported LAN/Link configurations.

LAN component -----	Fileset(s) required -----
UNIX Domain sockets	BSDIPC-SOCKET
IP/TCP/UDP	BSDIPC-SOCKET NETINET NET NETTRACELOG [LAN or X.25 driver]
Netipc	BSDIPC-SOCKET NETIPC NETINET NET NETTRACELOG [LAN or X.25 driver]
DUX or LLA	LAN NETTRACELOG
Appletalk	BSDIPC-SOCKET NET LAN NETTRACELOG APPLTALK

III. The networking memory reservation system, which was originally created to guarantee sufficient memory for all connections established, is unsupported in release 8.0 since it is no longer needed. Therefore, the configuration parameters NETMEMINIT, NETMEMTHRESH, and NETMEMMAX are unsupported in release 8.0.

IV. The command syntax for “ifconfig” and “netstat” are changed in release 8.0.

- “ifconfig” The 4.3 BSD version of “ifconfig” is supported. Previously “ifconfig” also configured link level encapsulation (ieee, ether). For 8.0, a new command, “lanconfig”, will configure the link level encapsulation.
- “netstat” New options for netstat include “-[I interface]” for showing information for a specified interface and “-[-f address family]” for limiting statistics or address control block reports to those of specified family.

V. Object Code Compatibility: Applications which ran on pre-8.0 systems will generally run on release 8.0 without recompilation or modification. While a few networking system calls have been changed, the older system calls are emulated on the 8.0 system.

VI. Source Code Compatibility: With only a few exceptions, any program which runs on an earlier HP-UX system can be run on release 8.0 without modification or recompilation. All Netipc networking system calls are both syntactically and semantically identical with earlier releases. No modifications will be required to recompile and run these applications on release 8.0.

All BSD networking system calls are syntactically identical except for “setsockopt()” calls and “ioctl(SIOCSPGRP)” call, as listed above. Programs which do not use the “setsockopt()” calls and Unix domain socket with path name greater than 92 bytes in length should require no modification to correctly compile and run on release 8.0.

VII. Changes in LLA functionality:

- The LLA NETCTRL network I/O control command will no longer support the ability to reset individual statistical counters.
- The “ioctl()” (LOG_READ_CACHE) system call is enhanced to set up caching for up to 64 additional packets for super-user processes and 16 additional packets for normal-user processes. This “ioctl()” supported up to 4 additional packets in previous release.

2-16 Part 1: Products That Relate to Filesets

VIII. Release 8.0 includes several new features for the LAN Link product:

- PPL (Point-to-Point Serial Networking) extends the Internet Protocol (IP) network over serial lines. PPL allows a user to directly establish a TCP/IP LAN connection over the serial line or indirectly through a remote dial-up to the HP 9000 serial IP line. The encapsulation protocols, SLIP (Serial Line Internet Protocol) and ASLIP (Abbreviated Serial Line Internet Protocol) are available.

PPL is supported over the following ARPA/Berkley services: “rlogin”, “rcp”, “telnet”, “remsh”, and “ftp”.

- Autoconfiguration is a feature intended for HP-UX release 8.0 on the HP 9000 series 800 SPUs.

Autoconfiguration allows the user of an HP 9000 Series 800 to modify the system's I/O configuration without modifying the kernel because the system “probes” to find out about the physical configuration. This implies that at 8.0 a user can change an I/O card (e.g. CIO LAN, NIO LAN, MUX) from one slot to another or add another I/O card or boot the same kernel on another processor (e.g. 855 or 835), all without reconfiguring the uxgen file and re-genning a new kernel. The user will however have to power down the processor to change I/O cards.

- New commands for release 8.0 are listed and briefly described below. For detailed descriptions, please see the corresponding man page.
 - “subnetconfig” Depending on the setting of the internal flag, each subnet of a network is viewed as either being a distinct network or belonging to the same network for the purpose of choosing maximum size of outbound TCP packets.
 - “lanscan” For each LAN interface, “lanscan” reports the hardware path, station address, device lu, hardware state, interface name and link type.
 - “arp” This program displays and modifies the Internet-to-Ethernet address translation tables used by the Address Resolution Protocol.
 - “lanconfig” This command configures the link level encapsulation (e.g. ieee, ether)

IX. Network Tracing and Logging changes.

At 8.0, a new facility for tracing and logging has been adopted. This facility replaces and obsoletes the following commands:

```
netlogstart(1m)
netlogstop(1m)
netlogfmt(1m)
nettrstrt(1m)
nettrstop(1m)
nettrfmt(1m)
```

The new commands are:

```
nettl(1m)
netfmt(1m)
```

The man pages for these commands should be consulted during the following discussion. The new facility combines the tracing and logging functions into a single subsystem. The previous command functions are implemented via the new commands as follows:

- “nettrstart(1m),” “nettrstop(1m),” “netlgstrt(1m),” and “netlgstop(1m)” are now obsolete. The “nettl(1m)” command must be used to start and stop tracing and logging.
- “nettrfmt(1m)” and “netlogfmt(1m)” are now obsolete. The “netfmt(1m)” command replaces their functionality.

Old Command	New Command
-----	-----
nettrstart	nettl -tracoon ...
nettrstop	nettl -traceoff ...
nettrfmt	netfmt ...
nettrdump	There is no analog for this command.
netlgstart	nettl -start
netlgstop	nettl -stop
netlgfmt	netfmt ...

CAUTION: there is no way to turn the logging facility off while still keeping the tracing facility available. Therefore, it is strongly recommended that the “nettl -start” command be executed during boot up, and the “nettl -stop” command should not be executed under normal circumstances. As long as the nettl subsystem is executing, logging can never be completely turned off, since

“Disaster”-level log messages are always recorded. However, all other levels of log messages can be turned off without shutting down the “nettl” subsystem.

Some previous options have different meanings or no analog in the new facility. These include:

■ **netlogstart**

-c mask and -C console

Console logging is enabled or disabled in the configuration file for “nettl”, “/etc/conf/nettlgen.conf”. See the “nettlgen.conf(4)” man page for a description of this entry. Only “Disaster”-class log messages go to the console. There is no way to specify another console device.

-f filename

The log file is also set in the “nettlgen.conf(4)” configuration file. See the man page for details.

-b (binary records)

The data is always recorded in binary mode. The “netfmt(1m)” command must be used to format the log data.

-d (run as daemon process)

“Nettl(1m)” starts up two daemon processes, “ntl_reader” and “nctl_daemon”. There is no provision to run these as foreground processes.

-m mask

The default logging classes are set in the configuration file; see “nettlgen.conf(4)” if you wish to change them. The logging classes can be changed using subsequent calls to “nettl” using the “-log” option. There are only four classes in the new facility. Resource Limitation events map onto Error class, while Logging and Protocol-Specific Information events map onto Informative class.

-B queue-size

The logging queue size is set in the configuration file; see “nettlgen.conf(4)” to change it.

-o optionfile and -U

There is no pre-filtering available on specific events. The same file used for pre-filtering prior to 8.0 can be used in the formatter, netfmt(1m), to post-filter the data.

■ **nettrstop**

-tr

This is implemented via the trace kinds “pduout” and “pduin” respectively.

-l

This is implemented via the entity “ns_ls_loopback”.

-b size

This is implemented via the “-size” option on “nettl(1m).”

-S subsystem...

This is not implemented. Use post-filtering available in “netfmt(1m)” to separate traces from the different network interfaces.

-p (preserve kernel buffer)

There are no kernel buffers used in the new trace facility.

Some of the items above mention that certain parameters can only be changed in a configuration file. To reconfigure the “nettl” and “netfmt” commands, a special script has been provided called “nettlgen(1m)”. This script rebuilds the “nettl(1m)” and “netfmt(1m)” programs using the data in the configuration file, “nettlgen.conf(4)”. See the “nettlgen(1m),” “nettlconf(1m)” and “nettlgen.conf(4)” man pages for more information.

_NCSNCK-RUN Release 1.5.1

This release contains Version 1.5.1 of the Network Computing Kernel (NCK). NCK, a component of the Network Computing System (NCS), provides runtime support for distributed applications. The other component of NCS, the Network Interface Definition Language (NIDL) Compiler, is a tool for developing distributed applications.

This release document describes NCK 1.5.1 for HP-UX 8.0 on HP9000 platforms.

1.1 Technical Changes in Version 1.5.1

This section summarizes changes made to NCK Version 1.5.1 since the HP-UX 7.0 release, as well as changes made since the previous version of NCK, Version 1.1. (NCK Version 1.1 was never released on HP-UX.) See the books listed in Chapter 2 for detailed information.

1.1.1 Global Location Broker Daemon

This release contains “glbd”, the replicatable version of the Global Location Broker (GLB) daemon. Previous releases of NCK for HP-UX only provided “nrglbd”, the non-replicatable version of the daemon. Because of the advantages of using a replicated database of RPC interfaces, we recommend that you use “glbd” rather than “nrglbd”.

When communicating with each other to maintain replication of the GLB database, GLB replicas should use only one protocol family. The Domain/OS glbd supports both the DARPA IP and Domain DDS network protocols. The HP-UX glbd only supports the DARPA IP network protocol.

You choose which protocol family the GLBs will use via a command-line option. In an internet, all routing nodes must support this family. If an HP-UX “glbd” is used in a network with Domain/OS GLBD replicas, then the Domain “glbd” replicas must also use DARPA IP network protocols for replication.

The following options for “glbd” have been added since Version 1.1:

- The “-family” option (added at Version 1.5) allows you to specify the address family of the first GLB replica you create. Replicas you create subsequently will use the same address family.

- The “-change_family” option (added at Version 1.5) allows you to change the address family of an existing GLB replica.
- The “-listen” option (added at Version 1.5.1) allows you to restrict the address families on which a GLB listens.

Managing NCS Software describes Location Broker configuration and the “glbd” options. A reference page for “glbd” is available online.

1.1.2 The “glb_obj.txt” File

The GLB is an object identified by a Universal Unique Identifier (UUID). This UUID has a default value. The “glb_obj.txt” file, new at Version 1.5.1, allows you to override the default value by specifying a different GLB object UUID for a particular host. This file supports special configurations that require several disjoint GLB databases (each of which is possibly replicated). In most networks and internets, there is only one GLB database (possibly replicated), and hosts do not need to have a “glb_obj.txt” file.

Managing NCS Software describes Location Broker configuration and the “glb_obj.txt” file. A reference page for “glb_obj.txt” is available online.

1.1.3 The “glb_site.txt” File

The “glb_site.txt” file, new at Version 1.5.1, lists the network addresses of hosts where a GLB daemon may be running. This file supports configurations where hosts are unable to locate a GLB via the usual broadcast mechanism.

Managing NCS Software describes Location Broker configuration and the “glb_site.txt” file. A reference page for “glb_site.txt” is available online.

1.1.4 The drm_admin Tool

As of Version 1.5, the Data Replication Manager (DRM) administrative tool, “drm_admin”, supports a “chrep” command that changes the address by which a replica appears in a replica list. This command supersedes the “reprep” command.

Managing NCS Software describes Location Broker administration and the “drm_admin” tool. A reference page for “drm_admin” is available online.

1.1.5 The lb_admin Tool

As of Version 1.5, the Location Broker administrative tool, “lb_admin”, can use textual names as well as UUIDs (Universal Unique Identifiers) to identify

2-22 Part 1: Products That Relate to Filesets

objects, types, and interfaces. In both the graphical and the command-line interfaces of “lb_admin”, textual names are accepted as input and displayed as output instead of UUIDs whenever possible. The associations of names with UUIDs are defined in the file “uuidname.txt”.

The following are new lb_admin commands:

- The set_timeout command adjusts the timeout period used by “lb_admin”.
- The clean command allows you to find and delete obsolete entries in a Location Broker database.

The “use_broker” command has been modified so that when invoked without an argument, it tells whether the Location Broker you’re dealing with is local or global.

Managing NCS Software describes Location Broker administration, the “lb_admin” tool, and the “uuidname.txt” file. A reference page for “lb_admin” is available online.

1.1.6 The “stcode” Utility

NCK now includes the “stcode” utility, which translates hexadecimal status codes to textual messages.

A reference description for stcode is available online and in the NCS Reference Manual.

1.1.7 The “uuidname.txt” File

The “uuidname.txt” file, new at Version 1.5, associates textual names with UUIDs. The “lb_admin” administrative tool can use these names to identify objects, types, and interfaces.

This mechanism for associating names with UUIDs may be superseded by a more general naming service in a future software release.

Managing NCS Software describes the “uuidname.txt” file. A reference page for “uuidname.txt” is available online.

1.1.8 The -version Option

As of Version 1.5, each of the NCK daemons and utilities accepts a -version option, which causes the command to print the version of NCK that it belongs to and then exit.

1.1.9 The System idl Directory

As of Version 1.5.1, the NCK product includes the system idl directory (/usr/include/idl), which contains NCS interface definitions and header files. This directory is identical to the one included in Version 1.5.1 of the NIDL product. and is included in the NCK fileset so that NCS-based applications consisting of C-language modules and NIDL- generated stub code can be compiled and linked on systems that do not have the Version 1.5.1 NIDL product installed.

The NCS Reference Manual describes the system idl directory.

1.1.10 Changes to Calls

The “rpc_\$inq_object” call now can be invoked either by a client or by a server.

The “rpc_\$name_to_sockaddr” and “rpc_\$sockaddr_to_name” calls are obsolete and have been superseded by the “socket_\$from_name” and “socket_\$to_name” calls.

The NCS Reference Manual describes the “rpc_\$” and “socket_\$” calls at Version 1.5.1.

1.2 Compatibility

If you plan to interoperate this version of NCK with NCK on Apollo Domain/OS system, you should read this section.

1.2.1 Versions of NCK

We have previously released NCK 1.5.1 as a layered software product for HP-UX 7.0. NCK software is part of standard operating system software in all Domain/OS Software Releases since SR9.7. The following list shows the version of NCK contained in each of these software releases.

SR9.7	NCK Version 1.1
SR10.1	NCK Version 1.5
SR10.2	NCK Version 1.5.1

All versions of NCK after Version 1.0 are mutually compatible. Version 1.0 is not compatible with any later versions.

1.3 Organization of this Release Document

- Chapter 2 lists the documentation available for NCK. and contains up-to-date additions, changes, and corrections to the documentation.
- Chapter 3 lists bugs and bug fixes.
- Chapter 4 contains information about installing NCK.
- Chapter 5 describes how to test and troubleshoot NCK once you have installed it.

1.4 Typographical Conventions

In syntax descriptions in this document, square brackets ([]) surround optional items, and ellipsis dots (...) indicate optional repetition of the preceding item. In procedures and examples, we use the dollar sign (\$) as a generic shell prompt.

Chapter 2: Documentation

This chapter describes documentation available for NCK Version 1.5.1. In general, documentation for NIDL and for the programmatic interfaces to NCK is provided with the NIDL products, while documentation for the runtime administration of NCK is provided with the NCK products.

2.1 Books

The following books are of interest:

- **Managing NCS Software.** HP Part number 011895-A01. This book, formerly titled *Managing the NCS Location Broker*, describes how to configure and administer the Location Broker and also discusses concepts of Location Broker operation. If you are running any software based on NCS, you should use the procedures in this book to set up the Location Broker. Appendixes of *Managing NCS Software* provide information about managing NCS software on particular systems.
- **Network Computing System Reference Manual.** HP order number 010200-A00, ISBN 0-13-617085-4. This book introduces NCS concepts, describes the “error_”\$, “lb_”\$, “pfm_”\$, “pgm_”\$, “rpc_”\$, “rrpc_”\$, “socket_”\$, and “uuid_”\$ calls, describes NIDL syntax, and describes how to develop distributed applications based on NCS. Appendixes of the NCS

Reference Manual provide information about using NCS on particular systems.

- Network Computing Architecture. HP order number 010201-A01, ISBN 0-13-611674-4. This book contains technical specifications for the Network Computing Architecture, including a formal definition of NIDL grammar.
- Network Computing System Tutorial HP order number 018355-A00, ISBN 0-13-617242-3. This book is a step-by-step guide to developing distributed applications using NCS.

2.2 Online Documentation

Fileset NCSNCK-MAN includes online reference descriptions (“man pages”) for the following software:

- NCK daemons and utilities
- NCK configuration files
- The “error_\$”, “pfm_\$”, and “pgm_\$” calls

Reference descriptions for other calls (for example, the “lb_\$”, “rpc_\$”, and “socket_\$” calls) are provided with the NIDL

Chapter 3: Bugs and Bug Fixes

This chapter describes bugs and bug fixes in the present release of NCK. See also the discussion of compatibility in Chapter 1 of this document.

Some bugs in the programmatic interfaces to NCK are described in the release documents for the NIDL products.

3.1 Known Defects in NCK 1.5.1 for HP-UX

There are currently no known defects in NCK 1.5.1 on HP-UX 8.0.

3.2 Defects Fixed Since the NCK 1.5.1 HP-UX 7.0 Release

3.2.1 “lb_admin” Limitation Removed

The set command in “lb_admin” now allows you to specify the Global Location Broker to which you want to send commands.

3.2.2 Bug Affecting GLB Performance

A GLB replica periodically purges from its propagation queue any update records that have been propagated. A defect that sometimes resulted in updates or lookups that happened during such purges to time out with a “resource locked” message has been fixed in this release.

Chapter 5: Testing and Troubleshooting NCK

Describes how to test/troubleshoot NCK after you install it.

This NCK release provides exercisers called “perf” and “lb_test”. These exercisers reside in “/etc/ncs”. The “perf” exerciser occupies a subdirectory of this directory.

You may wish to use the “perf” and “lb_test” exercisers to troubleshoot NCK, especially if you’re interoperating with other versions of NCK, or releases running on other platforms. The last section in this chapter discusses troubleshooting techniques.

5.1 The perf Exerciser

The “perf” exerciser consists of client and server programs called “client” and “server” that test the functionality and measure the performance of the RPC runtime library with minimal use of the Location Broker.

The “perf” client and server programs run as separate processes, either on one host or on two different hosts. Of course, any host that runs either the client or the server must have NCK installed.

To use “perf”, first start the server, then use the client to run various NCK tests against that server, as we describe in the next two subsections.

5.1.1 Starting the perf Server

Any host that runs the “perf” server must also run “llbd”, the Local Location Broker daemon. Ensure that an “llbd” is running.

The “perf” server has the following syntax:

```
$ server max_calls family
```

The “max_calls” argument must always be 1 on HP-UX systems. The family argument specifies one or more address families for which the server should create sockets to listen on. On HP-UX, the only valid family is “ip”.

For testing NCK functionality, a typical invocation of the “perf” server is:

```
$ server 1 ip
```

which creates a “perf” server that listens for “perf” client tests on an DARPA IP socket.

5.1.2 Running the perf Client

The “perf” client consists of 11 tests, each of which exercises a particular feature of NCK. To test NCK functionality, invoke the “run_client” script, which runs all 11 of the “perf” tests. (Several of the tests are run twice, once with the idempotent operation attribute and once without.) The syntax for “run_client” is

```
$ run_client family:hostname
```

On HP-UX, the only valid family is “ip”.

See the following subsection for descriptions of the “perf” tests and how you can run them individually.

5.1.3 “perf” Client Syntax

Following are the individual perf client tests.

Test	Syntax
-----	-----
Null call	client 0 host passes calls/pass verify? idempotent?
Variable length input argument	client 1 host passes calls/pass verify? idempotent? nbytes
Variable length output argument	client 2 host passes calls/pass verify? idempotent? nbytes
Broadcast	client 3 family
Maybe	client 4 host
Broadcast/maybe	client 5 family
Floating point	client 6 host passes calls/pass verify? idempotent?
Unregistered interface	client 7 host
Forwarding	client 8 host global?
Exception	client 9 host
Slow call seconds	client 10 host passes calls/pass verify? idempotent?
Shutdown	client 11 host

Test 0 makes the simplest possible remote procedure call, one with no parameters.

Tests 1 and 2 pass open arrays as input and output parameters, respectively.

Tests 3, 4, and 5 use the broadcast and maybe operation attributes. (Since broadcast ip packets are not generally routed by gateways, tests 3 and 5 will not work if the server and the client are running on hosts in different networks.)

Test 6 passes floating-point parameters.

Test 7 requests an operation in an interface that is not registered with the RPC runtime library at the server.

Test 8 sends a call to the Location Broker forwarding port at the server, testing the forwarding facility provided by llbd. (Test 8 requires an llbd to be running at the server. The others specify well-known ports.)

Test 9 makes a call that raises an exception at the server.

Test 10 makes a call from which the server will return after the specified period of time.

Test 11 uses the “rrpc_\$shutdown” call to remotely shut down the server.

The rest of this section describes the variables that appear in the syntax table.

- “host” is the host on which the “perf” server is running, specified in the form “family:name” (for example, “ip:toscanini” or “dds://szell”).
- “passes” is the number of times to run the test. The “perf” client displays a message after each pass.
- “calls/pass” is the number of remote calls to make during one pass.
- “verify?” is yes/no (y or n) input. A “yes” input to this field directs perf to run the test for correct operation. A “no” input directs “perf” to run the test to obtain performance statistics.
- “idempotent?” is yes/no (y or n) input. A “yes” input to this field directs the perf client to call an idempotent operation at the perf server. A “no” input directs the “perf” client to call a non-idempotent operation.
- “nbytes” is the number of bytes the input and output parameters should have.

- “family” is an address family (for example, “ip” or “dds”).
- “global?” is yes/no (y or n) input. At this release, specify n.
- seconds is the number of seconds the server should wait before returning.

5.2 The lb_test Exerciser

The “lb_test” exerciser tests Global Location Broker (GLB) functionality by repeatedly registering and unregistering entries with the GLB. It will fail if the GLB does not exist, if it is an incorrect version, if it is inaccessible, or if it is functioning improperly.

A Location Broker entry contains Universal Unique Identifiers (UUIDs) for an object, the type of the object, and an interface to the object, together with the location of a server exporting the interface. To avoid flooding the GLB database, “lb_test” always uses the same three UUIDs for all registrations.

5.2.1 Running lb_test

The syntax for “lb_test” is

```
$ lb_test [-unreg] [-num entries] [-time seconds]
```

The program loops, making 256 passes. Each iteration of the loop registers a set of entries, reads them back, then unregisters them.

- “num” specifies how many entries to register. The default for entries is 5.
- “time” specifies how many seconds the program sleeps between iterations. The default is to not wait at all.
- unreg causes “lb_test” to unregister all entries with the “lb_test” type UUID. After unregistering the entries, “lb_test” terminates.

To test GLB functionality after you have installed NCK, invoke “lb_test” with no arguments to use the default values for entries and seconds.

5.3 Troubleshooting NCK

This section gives some troubleshooting information for NCK.

If you suspect that NCK is not working properly, you should run the “perf” and “lb_test” exercisers as described in the first two sections of this chapter. The following hints may be useful:

- If the NCK exercisers are not running successfully, check that the networking software and hardware used by NCK is functioning correctly. On networks that support IP protocols, for example, try running “telnet”, “ftp”, and other IP-based applications between the hosts that ran the NCK exercisers. Network services must be available and working correctly before you start Location Brokers and other NCS-based programs.
- If the perf tests appear to be running successfully but NCK appears to be functioning incorrectly, try reversing the roles of the hosts. It is possible that the perf tests will work correctly with the client running on host A and the server running on host B but will fail with the client running on host B and the server running on host A.
- If any of the perf tests fail, try running the tests locally, with both the client and the server running on the same host. Failure in this case could be due to incorrect mapping of names to addresses (in a host table, for instance), absence of required daemons or servers, or mismatch of versions between the perf client and server.
- If lb_test fails, try running the lb_admin tool to access a Global Location Broker.

_Network Services

RFA is obsolete as of release 8.0.

_NFS/9000

NFS/9000 HP-UX File System

Support for a Disk Quota system is supported in HP-UX 8.0. In order to provide disk quota information for file systems mounted via NFS, the “rpc.rquotad” daemon is provided. The “rpc.rquotad” daemon, which is started by “inetd”, responds to requests sent by the “quota”(1m) command.

NFS/9000 Performance Features and Tuning

Several performance related enhancements have been added to the Network Information Service (formerly called Yellow Pages). The name “Yellow Pages” is a registered trademark in the United Kingdom of British Telecommunications plc. Most references to YP, yp, Yellow Pages and yellow pages have been changed to NIS, nis, Network Information Service and network information service respectively.

Performance Improvements

■ yppush

- “-m” and “-t” options were added to “yppush” to control the number of simultaneous yp map transfers (from master to slave servers), resulting in 3-5 fold improvement in map transfer rate.

- **ypserv**

- A defect was fixed to improve map transfer rate (up to a 3 fold increase).
- Server caching was added to improve server response to client requests. The size of the cache can be controlled with the “-c num” option where “num” is the maximum number of entries to cache for each map. The default value of “num” is 128. To turn off caching, start “ypserv” with the option “-c 0”.

NIS-maps

A few maps have been added to make library calls faster with NIS.

- “servi.bynp” for “getservbyname()”
- “rpc.byna” for “getrpcbyname()”

NFS/9000 HP-UX Daemons

To ensure proper operation on the 8.0 release of the HP-UX operating system, you may need to recompile applications that use the Remote Procedure Call (RPC) facility. For more information, see Appendix E of the “Installing and Administering NFS Services” manual.

_NLIO

Content

- Japanese NLIO
- Japanese Fonts
- Korean NLIO
- Korean Fonts
- Trad. Chinese NLIO
- Trad. Chinese Fonts
- Simp. Chinese NLIO

Dependencies

- X11
- Motif
- FA-FM
- Starbase

Impact

- Applications linking 7.0 kana-kan library can not use new new EUC feature.
8.0 kana-kan library must be linked to application for EUC.

New Features

- EUC compatibility for Japanese file codes
- User-defined character support in basic NLIO environment
- Extended user-dictionary support for Japanese host input conversion
- Korean Designer's font for Starbase Graphics Library
- Support for LaserJet II/III printer
- Support for the HP 700/X terminals
- Support for ITE in Japanese NLIO
- new size Japanese font for X11
- Enhancement on X11 based Japanese FEP

Deleted Features

- Support for X Window System Version 10 (X10)
- Support for 16-bit HP TextEdit widget
- Kanji Windex for Series 300

_NLS

NLS Enhancement: EUC/UJIS Support

The following high priority commands now support EUC/UJIS:

```
cat, cb, cc, cflow, cmp, comm, cp, cpio, cpp, csh,  
cxref, diff, diff3, echo, egrep, ex, fgrep, grep,  
lint, ln, lp, ls, make, man, mkdir, more, mt, mv,  
page, pwd, rm, rmdir, sh, tar, tcio, uniq, vi, view
```

NOTE: Other commands may work with EUC, but they are not tested nor modified for EUC. The behaviors are not guaranteed in 8.0. However, most commands which support HP-15 will work for EUC as far as the usage is limited to CS0 and CS1 characters.

The LANG environment variable needs to be set to "japanese.euc". This is a new language for EUC in 8.0. The "japanese" language is also available for HP-15 like in previous releases.

NLS Enhancement: ISO 8859-1 support

Support has been added for the ISO 8859-1 character set for Western Europe. The following locales are supported:

```
american.iso88591, c-french.iso88591, danish.iso88591,  
dutch.iso88591, english.iso88591, finnish.iso88591,  
french.iso88591, german.iso88591, icelandic.iso88591,  
italian.iso88591, norwegian.iso88591, portuguese.iso88591,  
spanish.iso88591, swedish.iso88591
```

The LANG environment variable, or appropriate LC_* environment variable, needs to be set to one of the names listed above to access the new locale.

NLS Enhancement: ISO 8859-2 ISO 8859-5 support

Support has been added for the ISO 8859-2 character set and the ISO 8859-5 character set for Eastern Europe. The following locales are supported:

```
bulgarian, czech, polish, rumanian,  
russian, serbocroatian, slovene
```

The “LANG” environment variable, or appropriate “LC_*” environment variable, needs to be set to one of the names listed above to access the new locale.

NLS Tools



Buildlang(1M)

HP added the following two keywords for writing buildlang scripts in order to support the EUC encoding scheme:

```
code_scheme
cswidth
```

These do not affect the 7.0 supported buildlang scripts.

HP-UX Release 7.0 buildlang scripts can be used as the input for Release 8.0 “buildlang(1M)” without any modification.

Locale.def file

Although the buildlang script file itself is forward compatible, the locale.def file created by the “buildlang(1M)” command is not compatible from Release 7.0 to 8.0.

All currently loaded language filesets must be updated.

All user generated (not HP supported) “locale.def” files built with the HP-UX Release 7.0 “buildlang(1M)” needs to be re-built by 8.0 “buildlang(1M)”.

JAPANESE fileset

The JAPANESE fileset includes the locale.def files for both JAPAN15 code set of the HP15 encoding scheme and the EUC Japanese code set of the EUC encoding scheme.

When installing 8.0, the customize script deletes the file:

```
/usr/lib/nls/japanese/ujis/locale.def
```

and the directory

```
/usr/lib/nls/japanese/ujis
```

from the 7.0 system and then creates the new directory

```
/usr/lib/nls/japanese/euc
```

and the file

```
/usr/lib/nls/japanese/euc/locale.def
```

for the 8.0 system.

Nl_tools_16(3C)

The “C_COLWIDTH” macro and “c_colwidth” function were added to write application programs that support both HP15 and EUC. The type of the return values of WCHAR and WCHARADV macros was changed from “int” to “wchar_t” (casted by unsigned int). These changes do not affect compatibility.

HP-UX Release 7.0 compiled objects which call “nl_tools_16” will run on 8.0 systems without re-compilation.

_Pascal

What's new for 8.0?

Command line options

-T produce a symbol table dump in the listing file.

Directives

- `include_search`—indicate where to search for include files
- `literal_alias`—preserve case of aliased variables and routines
- `shlib_version`—set version stamp in shared library module

Limited support for shared library modules—dynamic load library module production.

What's changed between 7.40 and 8.0?

- `new(anyptr)` not allowed
- better display of variant records when debugging
- new symbolic debugger information format
- upgraded Pascal reference manual for consistency and accuracy.
- preservation of file named in the program statement upon file closure.
- production of a warning when truncating input lines at 132 characters.

_SBDL

See the “_Starbase” topic.

_Software Dvmt Toolkit

Changes in Motif and X11 include files and library locations.

What

The include files for Motif and X11 are located in subdirs named “/usr/include/Motif1.1” and “/usr/include/X11R4”, and the libraries have been moved from “/usr/lib” to “/usr/lib/Motif1.1” and “/usr/lib/X11R4”.

Why

The Motif and X libraries are or will be shipped as shared libraries in HP-UX 8.0. Applications that use shared libraries incorporate an absolute path to the location of the shared library. The libraries have been relocated to subdirectories in “/usr/lib” so that applications may be run on a system that uses either Motif 1.0 or Motif 1.1, for example. This gives us flexibility in delivering Motif 1.2, X11 release 5, etc.

What you need to do

If you are building clients on 8.0 that use Motif 1.1 and/or X11R4, you need to specify the include path for the header files during compilation:

```
cc -I/usr/include/Motif1.1 -I/usr/include/X11R4 .....
```

During linking, you must specify the path for the libraries as well:

```
[cc | ld] -L /usr/lib/Motif1.1 -L /usr/lib/X11R4 .....
```

_Starbase

Starbase is a 2D/3D graphics library supporting bitmapped displays on the Series 300/400 and 800 family of computers. Starbase also supports graphics devices such as plotters, graphics terminals, and printers.

The filesets provided with Starbase are listed below:

FAFM-MAN	- Starbase Fast Alpha/Font Manager Manual Pages
FAFM-PRG	- Starbase Fast Alpha/Font Manager Programming Environment
FAFM-RUN	- Starbase Fast Alpha/Font Manager Execution Environment
FAFM-SHLIBS	- Starbase Fast Alpha/Font Manager Shared Libraries
SBDL-DEMO	- Starbase Display List Demos
SBDL-MAN	- Starbase Display List Manual Pages
SBDL-PRG	- Starbase Display List Programming Environment
SBDL-SHLIBS	- Starbase Display List Shared Libraries
STAR-DEMO	- Starbase Demos
STAR-MAN	- Starbase Manual Pages
STAR-PRG	- Starbase Programming Environment
STAR-RUN	- Starbase Execution Environment
STAR-SHLIBS	- Starbase Shared Libraries

With the 8.0 (s300) release, support was added for running Starbase on the following new bitmapped graphics subsystems:

HP98705
HPA1096
HPA1416
HP98736

The Starbase error catalogs (“/usr/lib/nls/american/sb.cat”, “/usr/lib/nls/n-computer/sb.cat”, and “/usr/lib/nls/C/sb.cat”) have been modified to include new error messages appropriate for new displays and functionality.

The asynchronous Starbase daemon includes support for the new displays.

The 8.0 release includes new Starbase calls for advanced rendering primitives supported by the HP98736 display. Also included are the new drivers for each of the new graphics subsystems, discussed under the section for the appropriate subsystem.

8.0 Starbase includes several defect fixes, including:

- Text color is now correct across appended text calls.

- You can now close a “dcpartial_polygon” with a “dccircle” without generating a core dump.
- A gopen to the overlay planes of the hp98730 driver with “RESET_DEVICE” no longer clears the color map.
- The 8.0 version of the sox11 driver works with HP X terminals and PCs. With previous versions of the driver, area fills with dithering and “block_write” render incorrectly, and “block_read” core dumps, when the graphics device is an X terminal or a PC running an X server with a 4-plane display.

Fast Alpha and Font Manager (FA/FM) are two libraries (libfa.a and libfontm.a) and related files which allow Starbase programs to write raster text on bit-mapped displays.

The FAFM-RUN fileset contains a set of raster font files for use by programs compiled with the FA/FM libraries.

The FAFM-PRG fileset contains the two libraries, plus include files, man pages, and demo program source code.

FA/FM may be used with any of the Starbase raster display drivers (“libdd98550.a”, etc). It may also be used with “libddsox11.a”, but in this case special actions must be taken as described below in the workarounds section.

HP-SBDL is an implementation of a software display list for use with Starbase. This release of HP-SBDL differs from the 7.0 release by the addition of the “dl_control(3g)” routine and enhanced memory management. As well, it is now included as part of Starbase.

The dl_control(3g) routine allows the user control of various display list operating parameters. It allows configuration of the new memory manager and greater control over picking operations.

Information concerning HP-SBDL may be obtained from the following manuals:

98674-90002 Starbase Display List Programmer’s Manual

The SBDL demos and examples have not changed for release 8.0.

2-42 Part 1: Products That Relate to Filesets

Starbase Support for HP98705 Display

1. Features

With the 8.0 release, drivers are provided to support Starbase graphics on the HP98705 display, both for raw device access and within X11 windows. The functionality of this display is very similar to the HP98731 (TSRX) display, with the following exceptions:

- No 24-plane configuration available (only 8- and 16-plane)
- No video blending
- No video pan and zoom
- 16 accelerated contexts supported rather than 32

Additional new features of this device include:

- Faster hardware cursor
- Hardware support for bit/pixel reads and writes
- Accelerated graphics contexts are now stored the graphics engine, to reduce swapping.

For further information on Starbase graphics on this device See the “HP 98704 Device Driver”, “HP 98705 Device Driver”, and “Using Starbase in X11 Windows” chapters of the “Starbase Device Drivers Library” manual (Part Number 98592-90025).

2. Drivers

These are the new drivers for the HP98705 display. Normal Starbase link sequence rules apply.

```
/usr/lib/libdd98704.a  
/usr/lib/libdd98705.a
```

Starbase Support for HP98736 Display

1. Features

HP98737A, HP98736A, and HP98735A graphics subsystems (hereafter collectively known as HP98736) are a high-end 3D graphics follow on to the HP98730 (TSRX) display. Many new features such as texture mapping, anti_aliasing, 12 bit indexing, 24 bit Z buffer, contouring and deformation, and model clipping and capping are supported.

HP98736 is not supported on series 800 systems. It is only supported on series 300 375 systems. All configurations come with 24 image planes, 4 overlay planes, 4 display mode/clipping planes, 24 bit Z buffer, and either one (HP98735A), two (HP98736A), or three (HP98737A) transform engines. Both HP98737A and HP98736A come with a Virtual Direct Memory Access (VDMA) interface card. HP98735A comes with a Physical Direct Memory Access (PDMA) interface card. X11 Windows is supported, but not X10 windows or HP/9000 windows. Up to 15 unique color maps per window are supported.

This family of new high-end 3-D graphics subsystems support many new features beyond those available on the 98730 subsystem. It is supported only on the HP 9000 Model 375 and the HP Apollo 9000 Model 400s systems.

In general, there is relink-and-run code compatibility between these new subsystems and code created for the 98730 for the X Window System, Starbase, and Starbase Display List. There are a number of differences, which may require changes in use of resources or modification of applications:

- Accelerated world coordinate lines may be displayed at slightly different locations (compared to their locations on the 98730).

Solution: Use the unaccelerated driver. This will provide pixel compatibility.

- Applications which employ a perspective transformation will have different values written to the Z buffer (as compared to previous graphics subsystems). Different pixels may be illuminated than on previous devices.

This will result in variations in perspective from those obtained on previous graphics subsystems.

- World coordinate accelerated polygons will be missing the top scan line of the polygon (compared to their appearance on the 98730).

2-44 Part 1: Products That Relate to Filesets

Solution: Use the unaccelerated driver. This will provide pixel compatibility.

- World coordinate accelerated moves and draws to the same point will not appear on the new graphics devices unless the coordinates happen to be at the center of a pixel. (Stroke text or markers are not affected.)

Solutions: Modify the application to use the “gescape” “DRAW_POINTS”. This allows the user to turn on or off the capability for the new graphics devices to draw points rendered as move/draw to one point or polyline to one point. Use of this gescape will degrade line-drawing performance slightly.

Another approach is to set the environment variable “HP98736_POINT_COMPATIBILITY_MODE”. This will allow the centering of points without modifying the program.

- The new graphics subsystems require that reads to and writes from the 24-bit Z buffer be word/pixel (rather than byte/pixel) operations. (Word length is 32 bits.) Byte/pixel operations performed on the Z buffer will result in segmentation violations (when reading) or erroneous data (when writing).

Solution: Modify the application to use word/pixel structures instead of byte/pixel. The 24 most significant bits of data are valid. The low 8 bits are not used when writing to or reading from the Z buffer.

- The PAN_AND_ZOOM gescape is not supported on the 98736.

Solution: The gescape PAN can be used to display a 1024x1024 region whose upper left corner is either “0,0” or “1024,0”. Limited pan and zoom support can be provided by GR2D_REPLICATE.

- The PATTERN_FILL gescape cannot use patterns larger than 32x32 pixels. Also, PATTERN_FILL must be called every time the pattern is changed.

Solution: Texture mapping may be used to implement two-operand patterns larger than 32x32 pixels. The application may have to be recoded to call PATTERN_FILL every time the user pattern is changed.

- The frame buffer of the new graphics subsystems permits direct access only for word/pixel operations. Direct access for byte/pixel operations is not allowed.

Solution: Modify applications to do word/pixel access to the frame buffer.

- When running Starbase in an X Window, only combined, overlay, and image mode are supported on the new graphics subsystems. Stacked mode is not supported.

Solution: Modify applications to use only combined, overlay, or image mode.

- The accelerated device driver for the new graphics subsystems cannot render to the Z buffer.

Applications which render to the Z buffer cannot be ported to the new graphics subsystems.

- Because the PAN_AND_ZOOM gescape is not supported on the new graphics subsystems, applications that require support for zooming cannot readily be ported to the new graphics subsystems.
- Blending via the gescapes IMAGE_BLEND and OVERLAY_BLEND are not supported on the new graphics subsystems.

Applications that require these gescapes cannot be ported to the new graphics subsystems.

Support has been added to Starbase for the HP98736 graphics subsystem. New functionality includes:

Texture Mapping

Support for this feature is implemented through the new Starbase entry points: "(bf_)texture_index", "define_texture", "texture_viewport", and "texture_window". Support for texture mapping is included in a set of new with_data primitives. A new interior style "INT_TEXTURE" is supported. Spline surfaces can also do texture mapping.

With data primitives

A new set of primitives have been created. These primitives support texture mapping, deformation, contouring, blending, and illumination factors. The new entry points are: "polygon_with_data", "polyhedron_with_data", "polymarker_with_data", "quadrilateral_mesh_with_data", and "triangular_strip_with_data".

Contouring

Contouring is supported on the “polyhedron_with_data” and “polygon_with_data3d” entry points. Three and four vertex polygons can be contoured in different styles. The new entry points that support contouring are: “contour_enable”, and “define_contour_table”.

Deformation

Deformation of vertices is supported in all “with_data” primitives. The new entry point for deformation is “deformation_mode”.

“Model clipping and capping”

Model clipping is the method of defining a set of clipping regions in the current model coordinate space which, when enabled, are applied to all subsequent primitives. Capping allows clipping information associated with a particular model clipping plane to be gathered and formed into a polygon which will cap a closed primitive volume clipped by the plane. Model clipping is supported through: “set_model_clip_indicator” and “set_model_clip_volume”. Capping is supported through “set_capping_planes”.

Anti-aliasing

Anti aliasing of lines is supported through the entry points “line_filter” and “perimeter_filter”.

Alpha transparency

Support for alpha blending of surfaces is given in “(bf_)alpha_transparency”.

12 bit indexing

A 4096 entry color map is supported. This means that in this mode, a 12 bit index is used. Modifications have been made to “double_buffer”, “define_color_table”, and “inquire_color_table” to support this mode.

2. Drivers

These are the new drivers for the HP98736 display. Normal Starbase link sequence rules apply.

```
/usr/lib/libdd98735.a  
/usr/lib/libdd98736.a
```

Starbase Support for the 98705A/B/C Graphics Subsystem

The 98705A/B/C is not supported on any series 800 system. On the series 300, it is supported on the following SPUs: 345, 360, 370, 375, and HP Apollo 9000 Model 400t, 400dl and 400s.

The 98705A/B/C is not supported in the DIO-I address space.

The 98705A/B/C is supported only by X11 Windows, whereas the 98721 and 98731 are supported for X11 Windows, X10 Windows, and HP Windows/9000.

The impact of the 98705A/B/C is best described in terms of the kinds of changes that will have to be made in applications designed to run on the 98731 and 98721 displays. Many 98721 and 98731 applications will be able to relink with the new drivers and run correctly with no source changes, provided they have followed the recommended practices for programming in a device-independent manner. Both accelerated (98705A/B/C) and unaccelerated (98704) drivers are provided, analogous to the hp98731 and hp98730 drivers for the 98731 device, with similar sets of capabilities. They are also analogous to the hp98720 and hp98721 drivers for the SRX, but have more capabilities. Applications that make use of the hp98720w driver should use the hp98704 driver when ported to the 98705A/B/C device.

There are certain features of the 98731 that are not available on the 98705A/B/C. Many of these are features used mainly by imaging applications.

- The frame buffer configurations available in the 98705A/B/C are 4 overlay planes plus either 8 or 16 image planes. No 24-plane configuration is available.

Applications that use 24-bit/pixel raster images will have to convert data to 8-bit/pixel in order to transfer it to the frame buffer. Some applications that rely on eight bits each for red, green, and blue banks, or that directly use 24-bit color indices, will also require code changes. However, all

2-48 Part 1: Products That Relate to Filesets

Starbase color map modes (CMAP_NORMAL, CMAP_MONOTONIC, and CMAP_FULL) are supported, and applications that use RGB triples to specify colors should work as expected. CMAP_FULL is supported on 98705A/B/C using a 3:3:2 subdivision of the eight planes. All advanced primitives, curved surfaces, shading operations, matrix operations, and light sources supported on the 98731 are also supported on the 98705A/B/C. A 16-plane Z-buffer is also included in the 16-plane frame-buffer configuration, and supports hidden-surface elimination in the usual manner. The 16-plane frame buffer configuration supports 8/8 double buffering in all color map modes.

- Video blending of the frame buffer banks is not supported. Applications that depend on the IMAGE_BLEND or OVERLAY_BLEND “gescape” will require changes.
- Pixel pan and zoom (via gescape PAN_AND_ZOOM) is not supported.
- The hardware cursor in the 98705A/B/C does not support rubber-band line and rubber-band box cursors. These are supported by both drivers in the fourth overlay cursor plane if it is available. If the application or environment is already using all four overlay planes for graphics, the unaccelerated driver can render rubber-band cursors by complementing the contents of the graphics planes, but the accelerated driver cannot support rubber-band cursors at all. This difference from 98731 support may require an accelerated application to place restrictions on its environment, or to use only three overlay planes instead of four, or to forego rubber band cursors.
- The hardware cursor colors in the 98705A/B/C are always either dominant over the image and overlay planes, or completely transparent. This differs from the 98731, in which the hardware cursor can be made non-dominant with respect to the overlay planes. The non-dominance allows a cursor belonging to an image plane to be visible only in regions of the screen where the image planes are visible.

In order to get this same effect on the 98705A/B/C, fourth overlay plane cursors must be used rather than the hardware cursor. The R_ECHO_CONTROL gescape can be used to enforce overlay plane cursors.

Only a small number of applications are expected to require this exact behavior in order to operate correctly; most can use the default (hardware

cursor), with a slight but non-detrimental difference in appearance to the user.

- There are only 16 accelerated states available on the 98705A/B/C, as opposed to 32 on the 98731. Each gopen requires one state, so applications requiring more than 16 simultaneous gopens with the accelerated driver (hp98705A) will be unable to operate. In a window environment, this limits the number of accelerated gopens by all applications being run simultaneously.

Here is a summary of the differences with possible source code impact between the 98721 and 98705A/B/C.

- Although four overlay planes are always present on the 98721, Starbase software only made use of three of them for graphics in the overlays, reserving the fourth for cursors. The color map for the overlay planes is very simple, with one bit each for red, green, and blue. The 98705A/B/C has a hardware cursor, and therefore it is often possible to open all four overlay planes for graphics, although the three-plane configuration is also supported. The number of planes to open is differentiated via the minor number of the device special file. The 98705A/B/C overlay color has a full eight bits of intensity control for each of red, green, and blue. Applications that use the 98721 overlay planes should port without changes to the three-plane configuration on 98705A/B/C, unless they are relying on the limited intensity of the 98721 when using RGB color specification.
- The 98721 is available with 8, 16, or 24 image planes, but the 98705A/B/C is only available in 8- and 16-plane configurations. The notes above about 24-plane applications on the 98731 also apply to applications on the 98721.
- Hidden surface elimination on the 98721 is supported using multiple rendering passes if there is not sufficient frame buffer memory to complete a rendering in one pass. On the 98705A/B/C, hidden surface elimination is only supported if the 16-plane Z-buffer option is installed. When this hardware is present, rendering can be completed in one pass, and the `hidden_surface(3g)` call should return a value of one. If `hidden_surface()` is called and the Z-buffer is not present, an error is reported and the return value is less than one. The `gescapes ZBUFFER_ALLOC`, `ZSTATE_SAVE`, `ZSTATE_RESTORE`, and `ZWRITE_ENABLE` have no effect on the 98705A/B/C.

2-50 Part 1: Products That Relate to Filesets

- The note above about dominance of the hardware cursor over the overlay planes applies to 98721 applications as well as 98731 programs. Normally the cursor dominance will not be a detriment to the user interface.
- The 98721 can support only one accelerated gopen (context) at a time. The 98705A/B/C supports 16 simultaneous accelerated contexts, so multiple gopens using the accelerated driver are allowed at the same time. In addition, the 98721 offers no support for accelerated graphics in windows, whereas the 98705A/B/C does perform the clipping and translation necessary for accelerated graphics in multiple windows. A common solution on the 98721 is to run the single accelerated rendering process in the image planes, and run human interface processes (text, other graphics, and perhaps a window system) in the overlay planes using the unaccelerated driver. The same scheme can be used on the 98705A/B/C for ease in application porting.
- There are backwards-compatible changes in the 98705A/B/C for some of the advanced rendering features available on the SRX. For example, 15 positional/directional light sources are available rather than eight, and 6th-order splines are supported rather than 4th. These kinds of changes should not require any source code changes when porting applications, but it may be desirable to take advantage of some of them.
- One difference that may require changes in some applications is the behavior of gescape LS_OVERFLOW_CONTROL, which handles light source calculation overflows. In the hp98721 driver, the HYBRID option is supported, whereas in the hp98705 driver, the surface_coefficients(3g) procedure must be used to achieve similar results, just as for the 98731.

There is also a new feature available in Starbase support for the 98705A/B/C that is not available on either the 98721 or 98731.

- The 98705A/B/C has hardware support for frame buffer reads and writes in bit/pixel mode. The Starbase driver makes use of this support in the implementation of “raw” mode block_read() and block_write() operations using the R_BIT_MODE and R_BIT_MASK gescape. The GR2D_PLANE_MASK gescape (previously usable only via the hp98550 and hp98556 drivers) is also supported. In addition, a new gescape, SET_REPLACEMENT_RULES, is provided to be using in combination with the bit/pixel block write capability to expand a single plane of data to all destination planes under user control. This feature should be useful to

applications that have single-plane raster images and need to expand them to one or two colors in all frame buffer planes.

Alternatives

Some alternatives for individual porting problems were mentioned above. Most applications will probably be able to be adapted to the new platform, many without source changes. If an application absolutely requires the blending, pan/zoom, or 24-plane features of the 98731, there is no alternative but to continue support on the 98731. The 98705A/B/C could not be used with such an application.

Starbase Support for HPA1096 Display

This is the new driver for the HPA1096.a. Starbase link sequence rules apply.

```
/usr/lib/libdda1096.a
```

1. Supplementary Programming Information

“Device File”

The HPA1096 board can be set to use one of 4 starting addresses. This is selectable by setting the jumpers on the board to 1 of 4 positions. The following table shows the correspondence of jumpers to select code.

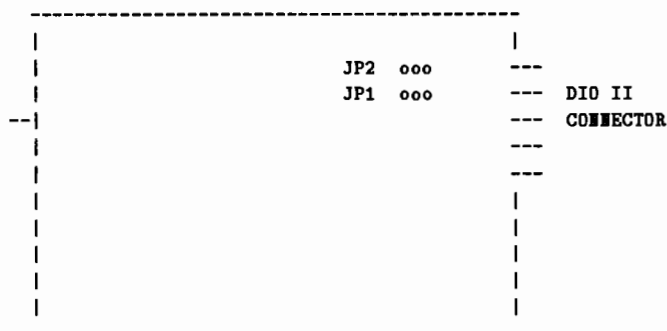
JP2	JP1	Select Code

0	0	132
0	1	133
1	0	134
1	1	135

The jumper consists of a shorting plug and three pins. The shorting connects two of the pins. Holding the board so that the video out connector is facing left and the DIO II bus connector is facing right, the 2 rightmost pins connected is a 0, the 2 leftmost pins connected is a 1.

2-52 Part 1: Products That Relate to Filesets

If this is the A1096A board:



When this jumper position is a 0: 0|00|

And this jumper position is a 1:

--
|oo|o

Starbase Support for HPA1416 (CH')

The A1416A color graphics board is supported on the HP Apollo 9000 Model 400t, 400dl and 400s only. It is entirely software compatible with the 98550 color graphics board.

The HPA1416 is both hardware and software compatible with the “libdd98550.a” driver. An example gopen statement would be as follows:

```
gopen("/dev/crt", O_RDWR, "hp98550", INIT)
```

NOTE: This display does not support the hp98556 accelerator.

Compatibility

To take advantage of 8.0 Starbase, the user must re-link his Starbase programs with the new Starbase libraries. See Starbase(3G) in the “Starbase Reference” manual (Part Number 98592-90067) as well as the “Using Starbase in X11 Windows” chapter of the “Starbase Graphics Techniques” manual (Part Number 98592-90080).

World coordinate moves and draws or polylines to the same point will not appear on the HP98736 device unless the coordinates happen to line up at the center of a pixel. This does not affect stroke text or markers. A gescape `DRAW_POINTS` will be supported which will allow the user to turn on or off the capability for the HP98736 display to draw points rendered as move/draw to one point or polyline to one point. The application will have to be modified to use this gescape. The gescape will cause slightly poorer performance for lines. Alternatively, an environment variable `HP98736_POINT_COMPATIBILITY_MODE` can be set to center points so that they are compatible with HP98730 displays.

HP98736 has no support for the `PAN_AND_ZOOM` gescape. The `PAN` gescape has been added to place the video display in a 1024x1024 output mode where the upper left hand corner of the video display is either at address 0,0 or address 1024,0. No zoom support is available. The `GR2D_REPLICATE` gescape can be used for limited zoom and pan functionality.

Blending via the gescapes `IMAGE_BLEND` and `OVERLAY_BLEND` are not supported on HP98736. Applications that use `IMAGE_BLEND` or `OVERLAY_BLEND` can not be ported to HP98736.

The `PATTERN_FILL` gescape on the HP98736 will not support patterns larger than 32x32. It must be called each time the user changes his pattern. Applications that use the `PATTERN_FILL` gescape can not use patterns larger than 32x32. If the application does not call `PATTERN_FILL` every time the user pattern is changed, the application will not function the same as on a HP98730.

Direct access to the frame buffer is word/pixel on HP98736 rather than byte/pixel. Applications that write the frame buffer directly will not be source code compatible with HP98730. Modify the application to do word/pixel access to the frame buffer.

Stacked mode for Starbase in X11 Windows will no longer be supported for HP98736. Combined, Overlay, and Image mode are still supported. Applications written to take advantage of Stacked mode can not run on HP98736 devices. Modify the application to use only Combined, Image, or Overlay mode.

2-54 Part 1: Products That Relate to Filesets

A line drawn in one direction between two points may not necessarily consist of the same pixels as a line drawn between the same points but in the opposite direction when using the “libdd9836.a” driver.

Workarounds

Direct frame buffer access with the “libdd9836.a” driver requires more than a call to R_GET_FRAME_BUFFER. After the frame buffer address is retrieved, a call to the PLUG_ACCELERATED_PIPELINE gescape is required before any direct frame buffer access. After this gescape, no Starbase calls will be processed until another call to the PLUG_ACCELERATED_PIPELINE gescape is made to unplug the pipe.

view_port and clip_depth are order dependent when using the “libdd9836.a” driver. Perform “view_port” before “clip_depth”.

If the “hp98720w” driver is used in raw Starbase mode, as opposed to in a Windows/9000 environment, FA/FM clipping should not be turned off. The “hp98720w” driver is primarily a Windows/9000 driver.

Using FA/FM with the SOX11 Driver

The X11 server now accesses fonts by font name rather than file name. One consequence of this is that programs which use “libfa.a” or “libfontm.a” with “libddsox11.a” will not work with 7.0 and later X11 servers unless you run the “fixfonts” script located in “/etc/newconfig” and modify your “.x11start” file as indicated in the output of the script. Run the “fixfonts” script as root.

Starbase Display List

The Starbase Display List product is bundled with Starbase at 8.0. You need not order it separately.

The Starbase Display List library has increased from 340 kilobytes (in HP-UX 7.0) to 500 kilobytes (in HP-UX 8.0).

Reconfigurable Operating Parameters

This new feature is implemented through “dl_control()”. It allows configuration of the new memory manager and greater control over picking operations.

Batched picking with the hp98736 and hp98705 Accelerated Drivers.

Pick_from_segment will use a batched-picking algorithm (which can result in increased picking throughput) under either of two conditions:

- if “PK_FIRST” is the current pick mode, or
- if “dl_control” is called with “SET_PK_SPEED” after any pick mode change.

The second case is a safe course of action in any pick mode, provided that the check_hit function does not rely on any Starbase or Starbase Display List state that can change during traversal, except for the pick path and pick path depth.

Enhanced Memory Manager

Starbase Display List has its own configurable memory manager. While the default configuration will be satisfactory for most applications, any application with special needs can call dl_control() to reconfigure it as required.

Correction to Rename Segment (and References)

In the HP-UX 7.0 release of Starbase Display List, execution of rename_segment or rename_segment_and_references caused the flags of a new segment to remain set if no old segment existed. Now, a new segment will always have no flags set following a rename when the old segment does not exist.

Any application that relies on the old behavior will have to be corrected to reflect this change.

Shared Libraries

At 8.0, Starbase will have shared libraries available. Note that both archived and shared libraries will be shipped with the product. There are three basic changes with the addition of shared libraries: the compile/link sequence is different, GOPEN has changed, and a new function INQUIRE_DEVICE_DRIVER had been added.

Advantages of the shared libraries are:

- No relinking for new hardware or next version of the software.
- Smaller compiled programs.

You cannot mix shared and archived libraries.

Compile/Link Sequence

Archive compile line:

```
cc -o prog prog.c -ldd98550 -lsb1 -lsb2
```

Shared compile line:

```
cc -o prog prog.c -lsb -ldld
```

GOPEN

The GOPEN function allows the driver to be specified as NULL. When you use NULL, at run time the program determines what device you have and dynamically links in the appropriate driver.

inquire_device_driver

This new function will return the appropriate driver if you are using shared libraries. If you are using archived libraries, the function returns the driver that was linked with the application.

inquire_capabilities

This new function will return information on the capabilities of the gopen'ed device. Capabilities returned include: zbuffer support, face culling, color mapping, lighting, shading, transparency, clipping, with data capabilities, anti-aliasing, NURBS, and per window d-buffering.

Starbase and ANSI C

KNR C is supported by Starbase. At the 8.0 release ANSI C is also supported. To use ANSI C, use a “-A” as one of your compile options.

Starbase Last Support for Some Software

8.0 is last release of some drivers and products. Some other sections on Starbase have information about support.

- 98700 and 98710 Starbase driver - last release
- 9836 Compatibility Card Starbase driver - last release
- Starbase Device Driver Development Guide - last release

Switchover/UX

Summary of Change

SwitchOver/UX is a new product in the Reliable On-Line Computing (ROC) product line for the Series 800. SwitchOver/UX provides near-continuous operation of systems by detecting system failures and allowing for a switchover between a primary and standby processor.

Configuration

SwitchOver/UX is an optional product (Number 92668A) supported on most Series 800 systems. It has two filesets, SWITCHOVER and SWITCHOVER-MAN.

Features

SwitchOver/UX supports up to 7 primary systems being backup up by a single standby system. In the event of a primary SPU failure, the standby SPU will take over the primary host.

SwitchOver/UX supports disk mirroring and multiple lan connections within host groups. It also provides disk locking to prevent data corruption during a takeover, and provides transparent network addresses such that the network address of the failed primary SPU can be carried over to the standby SPU.

Performance data

SwitchOver/UX has no significant effect on the core-system performance.

Size requirement

SwitchOver/UX requires 293k bytes of disk space and under 10k bytes of kernel memory. Additional kernel memory will be required if the SwitchOver/UX daemons are locked into memory.

_X.25/9000

New X.25 Installation Procedure

- 1) Configuration of X.25/9000 through SAM utility.

SAM can now be used to install, configure, initialize, remove, and shutdown X.25/9000. SAM replaces the X25install utility. It also can be used to execute X25init and X25stop for a X.25/9000 interface that has been configured. SAM can also be used to configure the new PAD products.

Changes for X.25 Programmatic Access

- 1) Semantics of “ioctl” to enable socket to receive SIGURG have changed

The semantics of the call to enable a socket to receive SIGURG using “ioctl(SIOCSPGRP)” have changed. Previously, if a negative process ID was passed to the “ioctl()”, it meant that the signal should be sent only to the process identified by that process ID. A positive process ID was interpreted as a process-group-ID and that the signal should be sent to the entire group with that process-group-ID.

In 8.0, the semantics are reversed: a negative process ID is interpreted as a process-group-ID and a positive process ID is interpreted as a single process ID.

- Impact: On programs that use out-of-band event signal handling. Note that this affects practically ALL X.25 user applications. Without any code change, the signals will not get delivered.
- Action: reverse the sign of process ID parameter (3rd parameter) to “ioctl(SIOCSPGRP)”.

Changes for Network Logging and Tracing for X.25

At 8.0, a new facility for tracing and logging has been adopted. This facility replaces and obsoletes the following commands:

```
netlogstart(1m)
netlogstop(1m)
netlogfmt(1m)
nettrstrt(1m)
nettrstop(1m)
nettrfmt(1m)
```

The new commands are:

```
nettl(1m)
netfmt(1m)
```

The man pages for these commands should be consulted during the following discussion. The new facility combines the tracing and logging functions into a single subsystem. The previous command functions are implemented via the new commands as follows:

- “nettrstart(1m)”, “nettrstop(1m)”, “netlgstrt(1m)”, and “netlgstop(1m)” are now obsolete. The “nettl(1m)” command must be used to start and stop tracing and logging.
- “nettrfmt(1m)” and “netlogfmt(1m)” are now obsolete. The “netfmt(1m)” command replaces their functionality.

Old Command	New Command
-----	-----
nettrstart	nettl -traceon ...
nettrstop	nettl -traceoff ...
nettrfmt	netfmt ...
nettrdump	(no analog for the nettrdump command)
netlgstart	nettl -start
netlgstop	nettl -stop
netlgfmt	netfmt ...

CAUTION: there is no way to turn the logging facility off while still keeping the tracing facility available. Therefore, it is strongly recommended that the “nettl -start” command be executed during boot up, and the “nettl -stop” command should not be executed under normal circumstances. As long as the nettl subsystem is executing, logging can never be completely turned off, since

Disaster-level log messages are always recorded. However, all other levels of log messages can be turned off without shutting down the nettl subsystem.

Changes for x29server

1) x29server access file formats have changed.

The format and the names of the “x29server” access files have changed for 8.0 to work with “x29uucpd”, “x29printd” and “padem”. “/etc/x25/x29access” does not exist anymore; it was replaced with “/etc/x25/x29hosts”.

“/etc/x25/convert_x29” converts the old “/etc/x25/x29access” to the new “/etc/x25/x29hosts” and convert “/etc/x25/convert_x3” converts the old “/etc/x25/x3config” to the new “/etc/x25/x3config”.

2) The “x29server” log file is under “/usr/adm/x29/x29server”.

The “x29server” log file is under “/usr/adm/x29/x29server” instead of “/etc/x25”. The server log file name is “x29server.log” and the child log file name is “x29logXXXXX” where XXXXX is the PID of the child.

3) “x29server” supports wild-card addressing.

The user may now use a wild-card character in the X.121 address to provide more flexibility in screening incoming PAD calls. The wild-card character is “?”. For example, a user may specify an address such as this: 40855512??. This will allow access to all incoming calls from remote PADs with addresses ranging from 4085551200 to 4085551299. A user MUST specify a question mark (?) for EACH digit in the address that should be a wild-card match.

4) “x29server” supports non-standard PID (protocol id) in calling packets.

X29server now accepts calls with non-standard PID's in the calling packets. Calling packets with 01 00 00 00 in the call user data field are accepted along with packets that have 01 XX XX XX in the call user data field. The only requirement is for the calling packets to start with 01 in its call user data field.

New Features in X.25 Programmatic Access

1) Capability to send X25 messages that greater than 64K (MDTF bit control).

This is a new feature of the X.25 programmatic access that allows an application to explicitly control the size of X.25 messages. The X.25 message is terminated by a packet with MDTF bit set to 0. In prior releases of X.25 link product, a complete X.25 message was formed for each send call. This effectively limited the maximum size of a message to 64 Kbytes.

In the current release, the application may string data from several “send()” calls into a single X.25 message. The state of the MDTF bit is changed by the user using “ioctl(X25_SEND_TYPE)” call. If a “send()” is issued at the time that MDTF is set to 1, a partial X.25 message is sent. When a “send()” is issued at the time that MDTF is set to 0, the X.25 message is completed.

2) Capability to receive X25 messages in user-specified fragments.

This is a new feature of the X.25 programmatic access that allows an application to receive partial X.25 messages. In prior releases, each “recv()” call resulted in reception of a complete X.25 message. This effectively limited the maximum size of receivable messages to 64 Kbytes.

In the current release, the application may explicitly control the size of partial X.25 inbound messages. This is accomplished by issuing the call “ioctl(X25_SET_FRAGMENT_SIZE)” giving the size of messages to be received. The application should also be checking the state of the inbound MDTF bit using “ioctl(X25_NEXT_MSG_STAT)” to determine when a complete X.25 message has been received.

3) Maximum number of vc's increased from 64 to 256 on S800 NIO machines.

The 8.0 release of X.25/S800 link product now supports up to 256 virtual circuits on each X.25 interface card in S800 NIO machines (e.g. S832, S815, etc). The S800 CIO X.25 card still supports only up to 64 virtual circuits per card.

New Features in IP-over-X.25

- 1) NFS support was added when used over X.25. This enables user to do NFS over X.25 but with read/write block sizes set to 1K.
- 2) Basic Network management support was added with basic MIB information stored. This identifies the X.25 system when used by MIB.
- 3) Default facilities are passed on to L3 on card even if user does not configure facilities for negotiation.
- 4) Set the DDN precedence to the configured precedence value if precedence is not set in Call Indication packet.

New X.25 Installation Procedure

- 1) Configuration of X.25/9000 through SAM utility.

SAM can now be used to install, configure, initialize, remove, and shutdown X.25/9000. SAM replaces the X25install utility. It also can be used to execute X25init and X25stop for a X.25/9000 interface that has been configured. SAM can also be used to configure the new PAD products.

New X.25 PAD products for 8.0

- padem: PAD emulation for X.25/9000 interface with PAD support (x29server). Padem is used to connect your local terminal with remote hosts over the X.25 network using the HP X.25/9000 interface.
- x29printd: PAD remote printer server for LP requests to remote printers on an X.25 network. X29printd is a server for remote printer support using the CCITT Recommendations X.3 and X.29 on an X.25 network.
- x29uucpd: PAD uucp server for uucp requests to remote hosts on an X.25 network. X29uucpd is a server to support uucp using the CCITT Recommendations X.3 and X.29 on an X.25 network.

New Command: "x25lbtest" on S300/S400 only

This is a new X.25 command used for diagnostic testing of the PDI card for the Series 300/400 machines. This command will perform loopback self test, testing just the PDI card, not the X.25 software.

2-64 Part 1: Products That Relate to Filesets

_X11 Windows Developer's Toolkit

Change in Motif and X11 include files and library locations

What

The include files for Motif and X11 are located in subdirs named “/usr/include/Motif1.1” and “/usr/include/X11R4”, and the libraries have been moved from “/usr/lib” to “/usr/lib/Motif1.1” and “/usr/lib/X11R4”.

Why

The Motif and X libraries are shipped as shared libraries in HP-UX 8.0. Applications that use shared libraries incorporate an absolute path to the location of the shared library. The libraries have been relocated to subdirectories in “/usr/lib” so that applications may be run on a system that uses either Motif 1.0 or Motif 1.1, for example.

What you need to do

If you are building clients on 8.0 that use Motif 1.1 and/or X11R4, you need to specify the include path for the header files during compilation:

```
cc -I/usr/include/Motif1.1 -I/usr/include/X11R4 .....
```

During linking, you must specify the path for the libraries as well:

```
[cc | ld] -L /usr/lib/Motif1.1 -L /usr/lib/X11R4 .....
```

_X11 Window System

The X11 Window System consists of three major components:

- HP VUE (Visual User Environment)
- OSF/Motif1.1
- X11 User Environment and Release 4 Server.

Visual User Environment (HP VUE)

HP-UX 8.0 includes HP's "Visual User Environment" (VUE), a powerful and intuitive graphical user interface built on OSF/Motif 1.1 and designed for greatly enhanced window-system usability. HP VUE offers secure graphical login features, expanded window management features, a graphical file/application management, a help system, and an interactive customizer for changing colors, fonts and other aspects of the user's environment. It also provides facilities for saving and restoring user sessions. The following section describes HP VUE components.

HP_VUE_Login_Manager

The HP VUE Login Manager provides a friendly, attractive and consistent means of initiating and terminating user sessions on both local displays and autonomous displays (e.g., X terminals). The session initiator is based on the "xdm" session initiation client from MIT and is compatible with the HP-UX login facility. Users who operate in (X) windows environments will log into the system using a HP VUE Login Manager screen. The HP VUE Login Manager will be installed by modifying "inittab" to replace "getty/login" entries. Specific features include:

- X Display Manager (xdm) functionality
- X Display Manager Control Protocol (XDMCP)
- User interface appearance and behavior based on R4/OSF/Motif
- Supports localization at login time (e.g., change the language)
- Access to a "no windows" environment (e.g., the ITE)

HP_VUE_Session_Manager

The HP VUE Session Manager (vuesession) is used to set up a user environment, save a user environment, lock a session (the user environment after login and prior to logout) and terminate a session. Vuesession provides the functionality previously provided by “xinit/x11start” and “x11lock” (a screen locking function).

HP_VUE_Workspace_Manager

The HP VUE Workspace Manager provides standard (“mwm”) window management and multiple virtual screens (workspaces) for window placement. The workspace manager (“vuewm”) features easy switching between workspaces and placement of client windows in one or more workspaces. Vuewm is compatible with “mwm” (“vuewm” and “mwm” are linked together) and will use existing “mwm” configuration files if started up as “mwm”. Specific features include:

- Supports new OSF/Motif 1.1 functionality
- User interface appearance and behavior based on R4/OSF/Motif 1.1
- Multiple workspace configuration used when run as “vuewm”
- Provides a front panel component for user environment management
- Closely integrated with other HP VUE clients (“vuehelp”, etc.)

The HP VUE File Manager provides a graphical view of the HPUNIX file system. It provides a direct manipulation paradigm (drag/drop) for the most common file manipulation operations: copy, move, remove, create a file or directory. Using a sophisticated file typing scheme, users can double click on executables and have them executed, or double click on a data file which will execute the application tied to that data file. Users can create new views of the same or multiple directories. Users can select to have their directories displayed with large icons, with small icons, simply as text, or in a directory graph. Data files can be dragged and dropped on executables which will then use the file as a parameter.

HP_VUE_Help_Manager

The Help Manager allows users to examine any and all help files which exist on the specified host. Help can be running on a host and yet getting its help files from some other host on the network.

Applications request the Help Manager to display the help contained within a data file and the Help Manager takes care of creating a OSF/Motif window in which to display and manage the help text. Users can also view and search for man pages within a system. Help contains a Help Browser/Index as well as a help display window. (An API is provided for application writers to make use of the HP VUE Help system, see entry on HP VUE API library.)

HP_VUE_Style_Manager

The Style Manager allows users to customize many aspects of their X/OSF/Motif environment graphically without hand editing default and resource files. Users can change colors, backgrounds, system font sizes, beeper tone, mouse click rate, screen timeout, password, hosts accessible, session save options, and keyboard choices.

HP_VUE_Utility_Clients

HP VUE Utility Clients are accessible from the front panel. HPTerm gives users a robust HP terminal emulator. Clock shows the current time via an analog clock face. Calendar displays the current date and day of the week. Load shows the cpu load. Printer prints a file using lp. Trash deletes a file or director dropped on it. Lock locks the screen from any input. Version displays the OS/Vue version numbers. The tools button lets users quickly open a file manager view on their tools directory. The home button provides a similar quick view of their home directory. Exit performs session save functions. Workspace switching, help browsing, ELM and style management are also accessible via attractive buttons on the front panel.

HP_VUE_API_Library_(libXv.a)

The HP VUE API library provides application writers with functions to access the help system (i.e., request the help manager to display help files for a particular application) and to save and restore client states across sessions.

OSF/Motif 1.1

The OSF/Motif 1.1 ToolKit replaces the HP-UX 7.0 OSF/Motif 1.0 Toolkit. OSF/Motif 1.1 migrates to X11R4 Intrinsics and provides Style Guide Conformance. This section specifies the changes from OSF/Motif 1.0 to OSF/Motif 1.1.

Change in OSF/Motif and X11 include files and library locations:

The include files for OSF/Motif 1.1 and X11 are located in subdirectories named “/usr/include/Motif1.1” and “/usr/include/X11R4”, and the libraries have been moved from “/usr/lib” to “/usr/lib/Motif1.1” and “/usr/lib/X11R4”.

The OSF/Motif and X libraries are shipped as shared libraries in HP-UX 8.0. Applications that use shared libraries incorporate an absolute path to the location of the shared library. The libraries have been relocated to subdirectories in “/usr/lib” so that applications may be run on a system that uses either Motif 1.0 or Motif 1.1, for example.

If you are building clients on 8.0 that use OSF/Motif 1.1 and/or X11R4, you need to specify the include path for the header files during compilation:

```
cc -I/usr/include/Motif1.1 -I/usr/include/X11R4 .....
```

During linking, you must specify the path for the libraries as well:

```
[cc | ld] -L /usr/lib/Motif1.1 -L /usr/lib/X11R4 .....
```

OSF/Motif_1.1_ToolKit_Content

The change X11R4 from R3 includes the following changes.

- Incorporates new R4 features which replace corresponding features added to OSF/Motif R3 Intrinsics (passive grabs, gadget focus, shell geometry)
- Adds complete support for multiple displays

- ICCCM compliance for Shells and selection manager
- OSF/Motif cut/paste code uses standard intrinsics selections
- Adds “layout if no resize” logic to “SetValuesAlmost” of Managers
- Arranging to use R4 Intrinsics implies:
 - Adds start-up hook by providing macro wrappers around Xt initialization functions
 - WM_Protocols and VendorShell code has been moved to Xm
 - Changes Position and Dimension to type short integer (see Resolution Independence section)
 - NULL is no longer a valid parameter for “XtAppContext” (see Appendix C, “X Toolkit Intrinsics, C Language Interface, X11 R4”)

Style_Guide_Conformance

The following changes were made to conform to the January 15,1990 version of the OSF/Motif Style Guide:

- move and display the location cursor with the focus.
- provides support for non optional mouse and keyboard bindings.
- Widgets will automatically add themselves as tab groups based on the “XmNnavigationType” resource.

CascadeButton

In a horizontally oriented menu (but not a menubar), a down arrow is used for the cascade graphic. A cascade button used as an option menu control is distinguished from a push button with a small horizontal bar on the right side of the button.

List Supports

Complete extended selection model (keyboard/mouse). Supports all relevant bindings.

2-70 Part 1: Products That Relate to Filesets

Text Supports

Primary paste, and using secondary selection, quick paste between widgets. Adds bindings for clipboard cut/paste. Supports complete extended selection model (keyboard/mouse).

ScrolledWindow

Provides actions for page increment and decrement, and for beginning and end of window.

Bulletin_Board

Moves the default button visual as the focus changes.

Buttons

Adds multiple-click capability via an “XmNmultiClick” resource.

Main_Window

Provides for command window in middle or bottom. Adds optional message area.

Paned_Window

Allows keyboard manipulation of sashes.

ScrollBar

Supports full bindings

Scale

Supports full bindings

Compound_string_and_Fontlist_Performance

Widgets with an “XmNfontList” resource will take their default from the closest BulletinBoard parent, or from the shell’s “XmNdefaultFontList” resource.

Widgets with an “XmNstringDirection” resource will take their defaults from their manager.

Resolution_Independence

Moves Conversion to set-values methods. Establishes Position and Dimension as type short integer.

ToolKit New_features

- Provides the ability to specify key bindings for keysyms which do not appear on a specific workstation. Because applications may run over a network, the bindings must be determined at runtime based on the location of the window server, not at compile time.
- Supports virtual key and button bindings, including bindings of the form:
 - KPageDown: PgDn -- binding to actual keysym
- Cleans up default color assignments. Provides better shadowing when background is very light or dark, very saturated or unsaturated. Provides better selection highlighting when background is dark.
- Provides version of XmGetPixmap that takes a visual as a parameter.

New_Convenience_functions

Adds convenience functions for workarea, menu, menubar, option menu, radio box (takes list of strings + callbacks, and for menus entries can also be separators and submenus).

Provides public functions for automatic color generation.

New Resources and Functions

Label_and_Buttons

Wider shadow in diamond toggle indicator. Adds XmNmenemonicCharset resource.

List Makes XmNdoubleClickInterval a dynamic default. Fixes visuals to make it possible to tell when the location cursor is on a selected item. Adds XmNtopItemPos resource Adds functions to return position of item, add/delete multiple items, and to delete multiple items by position.

Text Implement QueryGeometry

Does not ring the bell if a verification callback fails (or add a Boolean to the callback struct to prevent it)

XmNtopPosition has been changed to XmNtopCharacter due to conflict. Provides XmTextInsert(a,b,c) as a function equal to XmTextReplace(a,b,b,c).

Fifteen “hidden” functions have been made public.

Shells

Shells created by convenience routines are named after the widget which they parent by providing an appropriate suffix to its name. If XmNdialogTitle has not been specified for popup dialogue boxes, the name of the widget is used.

Supports both full and primary application modalities (See BB), and treats primary as full if window manager is not Mwm. Default positioning is deactivated upon user movement of window.

Manager

Adds an XmNnavigationGroup resource, with values XmGROUP_NONE and XmGROUP_TAB which determine whether or not the widget defines a tab group with appropriate defaults. Adds an XmNstringDirection resource.

ScrolledWindow

Fixes create convenience routines so that appropriate resources in the arg list are passed to the scrolled window, and not to the child.

OptionMenu

If menu is to be placed so that pointer is over default, and menu would go off screen, it is offset horizontally instead so that the pointer is off menu.

BulletinBoard

Activating a button from the keyboard activates it, not the default button if there is one. For XmNdialogStyle, defines the following values:

- XmDIALOG_FULL_APPLICATION_MODAL: locks out all windows in the application.
- XmDIALOG_PRIMARY_APPLICATION_MODAL: locks out all windows with same primary ancestor.
- XmDIALOG_APPLICATION_MODAL: remains as a deprecated synonym for XmDIALOG_PRIMARY_APPLICATION_MODAL.

Form Implements QueryGeometry.

Calculate initial size in ChangeManaged Adds a resource that enables a form to initially be the smallest size possible holding the children so that all constraints are satisfied. Form now draws a shadow whenever it has a non-zero shadow thickness.

FileSelectionBox

Horizontal scrolling is initialized so that the entry name is the leftmost character (not the two characters before it) Subdirectory traversal is added, requiring the addition of 9 new resources and caused another list to be displayed.

Primitive

Adds a XmNnavigationGroup resource, with values XmGROUP_NONE and XmGROUP_TAB, which determines whether the widget defines a tab group or not with appropriate defaults.

Gadget

Adds a XmNnavigationGroup resource, with values XmGROUP_NONE and XmGROUP_TAB which determines whether the widget defines a tab group or not with appropriate defaults.

Scrollbar

Provides improved visuals. Dynamically defaults to XmNprocessingDirection.

Scale

Provides improved visuals. Dynamically defaults.

XmNprocessingDirection

Supports non-pixel movement values. XmNscaleMultiple is a new resource added for scale movement.

New Widgets for Performance

One_line_TextField_Widget Single line text entry. Limited functions and resources.

Window_Manager_Contents

Specifications and changes for the OSF/Motif Window Manager, concern the following aspects:

- Reads “.motifbind” to set up virtual bindings property
- Provides support for both full and primary application modality.
- Adds support for managing multiple screens (with single mwm process)
- The button and keyboard bindings which apply to icons work, whether icons are on the root window or in the icon box.
- Adds a function f.raise_or_normalize which applies to icons. If the corresponding window is mapped, raises it; if not, normalizes (and raises) it.
- Adds a raiseKeyFocus resource. When set, and a mapped window is raised by invoking f.raise_or_normalize in explicit keyboard focus mode, the raised window gets the focus.

- Supports changes to the iconBoxTitle resource value.
- Adds an enableWarp resource which enables warping for keyboard triggered move and resize.
- Uses \$LANG to look up the .mwmrc file, and to determine the byte encoding to be used when reading it. Provides support for all character sets specified by the X Consortium for use with Compound Text.
- When icon box scrolling is not needed scrollbars are hidden.

Numerous manual pages associated with the OSF/Motif toolkit (Xm) have been improved for the 8.0 release.

X11 Server for the 8.0 Release

The X11 server, though fully backward compatible at the X protocol level with the 7.0 server, has undergone significant modification for the 8.0 release. The changes support new graphics displays, bring the server up to conformance with the MIT R4 release of X11, and improve performance. (The 7.0 X11 server conformed to the MIT R3 X11 release.) Most existing X11 clients (R3 or earlier) that operate properly with the 7.0 server should operate properly with the 8.0 server. Several clients however, relied on defects in the MIT R3 release for proper operation.

The changes made to the X Server for 8.0 were specifically to address support of new graphics displays (VRX, Color VRX, Personal VRX, Turbo VRX) and SPU hardware (Series 400), and to bring the server into conformance with the MIT R4 release. The X Server also includes scalable typeface support. Additional related changes were required in several supporting utilities (i.e. grmd, xwcreate, gwind) to support the Turbo VRX display family. Most changes described here for 8.0 were first incorporated in the HP-UX 7.03 Technical Core Release.

8.0 X11 Server Changes and Features

- Maintains full compatibility with the 7.0 X11 server
- Supports VRX, Color VRX, Personal VRX, and Turbo VRX graphics
- Supports Series 400 SPU's (68030 and 68040 versions)
- Supports access to scalable typeface outlines

2-76 Part 1: Products That Relate to Filesets

- Conforms to the MIT R4 release
 - Lower per-resource memory usage
 - Improved locality of reference in/among data structures
 - More correct/robust memory allocation error handling and reporting
 - Improved performance (particularly window operations)
 - Correct and faster wide line rendering
 - Support for standard input extension
 - Support for X Display Manager Control Protocol (XDMCP) extension
 - Support for connection authorization
 - Support for non-rectangular window extension
- Improved performance resulting from R4 conformance and display specific optimizations.
- Correctly operating SaveUnder support (enabled by default).
- Supports a shared memory client/server transport option for selected performance improvement of many operations.
- Numerous fixes for defects in the 7.0 server.

Backward Compatibility Mode

The MIT R4 server corrected many defects in the R3 server. Unfortunately, several R3 clients exploited some of the defects in R3, resulting in improper operation with an R4 server. To provide for backward compatibility, the MIT R4 server provides a command line option (bc) to select bug- for-bug compatibility with the R3 server. The server for the 8.0 release does not by default enable bug compatibility mode. If the user has an R3 or earlier client that appears to have problems operating with the 8.0 server, running the server in bc mode may help. The bc mode may be enabled at server startup by using the bc command- line option. See the server man page for details.

SaveUnders

The 8.0 server includes a correctly operating SaveUnder implementation. For 8.0, SaveUnders are enabled by default. SaveUnders may be disabled by starting the server with the `-su` command-line option.

Scalable Typeface Support

The X Window System now supports scalable typefaces through the existing X interface and is accessible through the standard X programmatic interface, Xlib. In most cases, this typographic functionality can be used with existing X applications, without requiring any change to the executable program.

The scalable typeface software is based on the Agfa Compugraphic Intellifont System and is compatible with the HP MasterType Library. This is the same scalable typeface technology used in PCL 5 printers.

HP enhancements include support for a large choice of character sets, artificial emboldening of typefaces, rotation of characters, obliquing fonts, and programmatic subsetting of character sets for improved performance.

The scalable typeface outlines included with the system release are CG times, Univers, and Courier. Each scalable typeface is provided in four treatments: normal, bold, italic, and bold italic.

The typefaces can be accessed through the standard XLFD names. For example you can view a 15 point Univers typeface by typing:

```
xld -fn "-cg-univers-*-*-*-*-*150-*-*-*-*-*"
```

You can query the X Window System for XLFD strings that describe available scalable outlines by typing:

```
xlsfonts | grep 0-0-0-0
```

Note that this also allows you to purchase scalable typefaces for the LaserJet family of printers and to use those scalable outlines in the X Window System. See the documentation for “`stmkfont`”, “`stmkdirs`”, “`stload`”, “`stlicense`”, and “`stconv`” for further details.

The Intellifont Scalable Typeface Library from Agfa Compugraphic includes more than 180 different designs. Forty-six individual volumes of 4 faces are available as well as six application-specific collections for all office publishing

2-78 Part 1: Products That Relate to Filesets

needs. Please call Agfa Compugraphic directly at 1-800-873-FONT (3668) for a free brochure or contact your local Hewlett-Packard Sales and/or Customer Service office for more information.

- “Intellifont” is a registered trademark of Agfa Corporation.
- Scalable type outlines are licensed from Agfa Corporation.
- “CG Times”, based on Times New Roman under license from the Monotype Corporation plc, is a product of Agfa Corporation.
- “Univers” is a registered trademark of Linotype AG and/or its subsidiaries.
- “Univers” and “Courier” are products of Agfa Corporation.

xwcreate and gwind Changes

Several changes were required to xwcreate and gwind to support the Turbo VRX display family. The man page for xwcreate has been updated as required:

- xwcreate changes
 - command line option “-visual” added
 - supports “VisualClass” X defaults specification of visual class
 - “man” page updated to reflect new -visual command line option
- gwind changes
 - utilizes visual type from xwcreate command line option if specified
 - validates window depth and visual class via separate calls to XgetVisualInfo()
 - returns visual class (in SoxCreateRec structure) instead of window class in response to request for window information (if NOTSTARBASE compile option is set)

Non-Rectangular Window (Shape) Extension Support

The 8.0 X11 server (and Xlib library) supports the non-rectangular window extension. This extension is supported for the first time with the MIT X11 R4 release and may exhibit certain anomalies in operation. These anomalies generally are cosmetic in nature and a simple refresh of the screen will correct them.

Standard Input Extension Support

The HP-UX 8.0 X11 server supports the Standard Input Extension. Documentation for this extension may be found in the on-line man pages distributed with the release. These man pages are not distributed in hard-copy form with the 8.0 release.

Incorrect rgb.txt Entries

The MIT X11 R4 release included changes to the “rgb.txt” color database. The HP-UX 8.0 version of “rgb.txt” has several incorrect entries containing underbars (_) in the color names. A user experiencing a problem with these names should edit the “rgb.txt” database file and recompile it into new “rgb.dir” and “rgb.pag” files using the “/usr/bin/X11/rgb” compiler utility. The incorrect entries will be fixed in all future HP-UX releases.

Shared Memory Transport Support for the X Server

With the 8.0 release of X11 a shared memory based transport facility for communication between X11 clients and the X11 server is supported. Since sockets can buffer only a small amount of data, a client is forced to do many socket calls to transfer a large protocol request to the server. Each time the client fills up the socket buffer it must pass control over to the server to empty it on the other side of the socket. This results in a significant amount of system overhead. The shared memory transport facility works by passing X11 protocol requests through a shared memory segment instead of sending them through a socket. With the shared memory transport facility a large protocol request is packed in its entirety into the shared memory segment in one operation. The server is activated only after the entire request is available. This removes much of the system overhead associated with the use of sockets and thus may greatly speed up the operation.

2-80 Part 1: Products That Relate to Filesets

Using the Shared Memory Transport Facility

These are the requirements that must be met to use the shared memory transport facility:

- The server must support the shared memory facility and the client must be compiled with an Xlib that supports the shared memory transport facility. The 8.0 X11 server supports the transport as does the 8.0 version of Xlib.
- The client must be run on the same machine as the server.
- The display parameter of the client or the DISPLAY environment variable must be set to “shmlink:{display number}.{screen number}”.

Advantages of the Shared Memory Transport Facility

Increased throughput is the main advantage of using the shared memory transport facility. Whenever an application needs to send a large amount of data to the X11 server via an Xlib call or via a long string of Xlib calls, significant gains in performance can be achieved. Examples of such calls are printing a very large amount of text, drawing a large number of lines, or sending a large image to the server with a PutImage call.

Disadvantages of the Shared Memory Transport Facility

There are several drawbacks to using the shared memory transport facility:

- A significant amount of system memory is consumed by the shared memory segments used by the shared memory transport facility. Each segment can potentially grow to a size of 512 Kb (though in typical operation, much less memory will be required). With one shared memory segment used per connection to the server, memory utilization may be a problem, offsetting the performance gains of using the transport.
- Performance gains will not be seen on all display types. The server may already be rendering images to a display as fast as possible using sockets. Increasing the rate at which data gets to the server does not mean that the server can render it any faster.
- Using many shared memory segments can tax operating system resources and actually cause server performance to degrade. It is because of this that it is

recommended that the shared memory transport facility not be used as the default means of connecting to the server.

X11 Window Documentation

The HP-UX Reference manual pages for X and the X server have undergone revision for the 8.0 release. They now reflect the R4 conformance of the server.

- HP VUE Programmer's Guide (new)
- HP VUE User's Guide (new)
- HP VUE System Administration Manual (new)
- HP VUE Configuration Guide (new)
- X Window System C Quick Reference Guide
- Programming with Xlib (R4 Version)
- X Toolkit Intrinsics Programming Manual (R4 Version) O'Reilly Volume 4
- X Toolkit Intrinsics Reference Manual (R4 Version) O'Reilly Volume 5
- Installing and Configuring the X Window System
- Using the X Window System
- HP OSF/Motif Programmer's Reference Manual
- OSF/Motif 1.1 Information Guide (new)
- HP OSF/Motif Programmer's Guide Prentice-Hall
- HP OSF/Motif Programmer's Reference Prentice-Hall
- HP OSF/Motif Style Guide Prentice-Hall
- Mastering Motif Widgets (new)

The 7.0 manuals "Beginner's Guide to the X Window System" is incorporated into the "HP VUE User's Guide". "Programming with Xrlib" and the X10 "Programming with the X Window System" are no longer available.

In addition to the manuals listed above, HP VUE will have an online help system that is context sensitive and provides access to online documentation.

All of the OSF/Motif documentation will reflect changes brought about by version R4 of the X Window System. The new manual “Mastering OSF/Motif Widgets” will explain many of the “hidden” features of widgets, such as dynamic defaults, interaction of resources, etc. It will do so through extensive use of small, easily understood programs.

PAM (Personal Applications Manager)

PAM is unchanged in the character-based terminal environment. However, PAM is no longer a “window smart” application. PAM now behaves identically in the X11 Windows environment and on a terminal. PAM no longer creates new windows for application programs executed from within PAM or via the VIEW key. Mouse input is limited to softkey availability only.

Release 8.0 is the final releases of PAM.

Part 2: Release-specific Topics

Please use the topics in this part for release-specific topics that relate more to the system or to HP than to a product the customer may purchase.

_Backup and Recovery of Files

fbackup(1M) and frecover(1M) changed to support MO and DAT media

Support for MO devices

MO (Magnetic Optical) devices provide about 325 Mbytes storage per side per disk. They are already supported by the HP-UX system. The C1701A 650A single disk drive provides one drive and one disk, which means 650 Mbytes storage. The C1701A 20GB autochanger provides two drives and 32 disks with autochanger mechanism, which means about 20 Gbytes total storage. These devices can be used as mounted file systems and as raw devices.

“fbackup(1M)” and “frecover(1M)” have been modified to support MO devices. They were modified to write to the next file or device when the current file or device becomes full.

Support for DAT devices

Digital Audio Tapes (DAT/RDAT) use the same device driver interface “mt(7)” with new features. Current programs that work with “mt(7)” should work with RDAT without changes. Because of its storage size, it will take up to 2 hours to recover a file from the tape.

RDAT provide a mechanism called “fast search”. The “mt(7)” driver now provide new operations (MTFSS, MTBSS) to use this capability. Frecover(ADMIN) will use the DAT fast search function to speed up the recovery of files.

fbackup(1M) and frecover(1M) specific changes

“frecover(1M)” can take multiple “-f” options to specify multiple files.

“fbackup(1M)” and “frecover(1M)” can use Pattern Matching Notation in the “-f” option to specify multiple files or device files easily.

The “fbackup(1M)” command has new options to give control over backing up symbolic links (“-s”), crossing NFS mounts (“-n”), the location of the database file for incremental backups (“-d”) and dumping the volume header (“-V”).

The “frecover(1)” command has a new option for printing out the volume header (“-V”).

Since network special files are no longer supported in 8.0 “fbackup(1M)” will not backup them up and “frecover(1M)” will not recover them. A warning message will be issued if a network special file is encountered.

fbackup(1M) and frecover(1M) compatibility

“frecover(1M)” on HP-UX 8.0 will be able to read all backup data made by “fbackup(1M)” on HP-UX 7.0 or before.

Old “frecover(1M)” may not be able to read backup data made by “fbackup(1M)” on HP-UX 8.0. Old operations will be accepted by “fbackup(1M)” and “frecover(1M).”

3-2 Part 2: Release-specific Topics

_Computer Languages

ANSI/C Compiler

For the Series 800, the following change was made.

User visible options for in-line compiler directive/pragma:

```
#pragma HP_DESTINATION ARCHITECTURE [1.0] [1.1]
#pragma HP_DESTINATION SCHEDULER [1.0] [1.1]
```

For the Series 300, the following changes were made.

OSF Data Alignment Pragas

The 8.0 Series 300 C compiler provides a new pragma (HP_ALIGN) to align structures to be compatible with the Series 500, Series 800, DOMAIN series 3000/4000, DOMAIN series 10000, and provide an alignment mode intended to be the default for OSF.

Instruction Scheduling

For the series 300 8.0 release c2 (the peephole optimizer) now performs instruction scheduling. The scheduler takes advantage of the MC68000's ability to concurrently execute floating point and integer instructions, and attempts to avoid sequences that cause A-register or FP-register interlocks which result in a pipeline stall.

New Debug Information Format

In the 8.0 release, the series 300 C compiler generates a new format for debugging information which is consistent with that of the series 800. This new format is inconsistent with prior Series 300 releases. Consequently, programs compiled with debug information must be recompiled to work correctly with the 8.0 debuggers.

\$ as Valid Identifier Character

To ease portability of C source from DOMAIN to HP-UX platforms, the 8.0 Series 300 C compiler now accepts "\$" as a valid identifier character (Note, not as the first character of an identifier). "\$"s are unconditionally accepted

in compatibility mode and conditionally accepted in ansi mode (the “+e” cc option enables its acceptance in ansi mode).

Procedure Integration (optimization level 3)

Available only with the C/ANSI C Product.

The S300 8.0 C compiler supports a new optimization level “+O3” which includes all of the level 2 optimizations, and in addition performs procedure integration. Procedure integration will replace certain function calls with the actual function code, which can increase performance by removing the call overhead and allowing more optimizations on the in-lined code. Several pragmas (HP_INLINE_LINES, HP_INLINE_FORCE, HP_INLINE_OMIT, HP_INLINE_DEFAULT, and HP_INLINE_NOCODE) have been added to provide finer control on procedure integration.

Support for Shared Libraries

The series 300 8.0 C compiler provides “+z” and “+Z” as options to generate position independent code (PIC). The C compiler also provides a new “HP_SHLIB_VERSION” pragma to associate a shared library version number with a module.

Fortran

Differences between 8.0 and 7.40 Fortran/300

I. NEW I/O LIBRARY

A new I/O library, compatible with the Series 700/800, is provided with this release. This library is the default I/O library. In addition to the compatibility across platforms, the key benefit of this library is that recursive I/O is permitted. For example, a WRITE statement can contain a function call that itself performs a WRITE.

The library that previously was the default is provided for compatibility. To use the compatibility library, use the +I300 option. The two I/O libraries are not compatible. Mixing calls to both libraries in the same program is likely to cause unpredictable results.

II. SHARED LIBRARIES SUPPORT

3-4 Part 2: Release-specific Topics

FORTRAN/300 supports shared libraries. Two options are provided for creating shared libraries from Fortran source files. A shared library version of each of the Fortran libraries is provided to support library calls (e.g. I/O, intrinsics) from user-created shared libraries.

III. POSIX-CONFORMING FORT77 COMMAND

The command “fort77” will provide a POSIX-conforming interface to the FORTRAN/300 compiler. When fort77 is used, two command-line options behave differently (-L, -N). One additional option is available when fort77 is used (-V).

IV. ALIGNMENT

By default, “REAL*8” and “COMPLEX*16” items outside of structures on the stack or in saved data are aligned on 8-byte boundaries. The “+A3” and “+A8” options cause “REAL*8” and “COMPLEX*16” outside of structures to be aligned on 8-byte boundaries. These changes should not be visible to a correctly written program.

V. STATIC ANALYSIS SUPPORT

The -y option provides support for Softbench static analysis. This feature was released with 7.40, but will first be available with SoftBench 8.0.

VI. MESSAGE CATALOGS

The compile-time messages issued by the driver and front end are in message catalogs.

VII. FCOPTS FEATURE

The “|” character in the FCOPTS environment variable causes options that appear after the “|” to be recognized after the command line options.

Pascal

For the Series 800, Pascal has been changed as follows. User visible options for in-line compiler directive/pragma:

```
$HP_DESTINATION ARCHITECTURE [1.0] [1.1] $  
$HP_DESTINATION SCHEDULER [1.0] [1.1] $
```

Symbolic Debugger, Series 300/400

Several new features are available with the 8.0 release of the HP Symbolic Debugger. The user is encouraged to examine the new manual:

HP Symbolic Debugger User's Guide (P/N B1699-90601)

Shared libraries

The most important new feature is the ability to debug programs that have been linked with shared-libraries. Although symbolic debugging of linked shared-libraries is not supported, the user has the ability to view shared-library code in disassembly mode, set or examine data associated with the library, or call shared-library functions from the command line. The linker symbol table for the library is automatically loaded by the debugger, so all address within the library may be referenced symbolically via a special syntax using "@" (e.g. "_printf@libc").

Breakpoints may not be set with shared-library functions, and the user cannot single-step into one; instead, the debugger steps "over" shared-library calls as if they were system calls.

ANSI/C and Debugger

Support for new ANSI/C types (long double, wchar_t) has been added since the 7.0 release of the debugger. These features have already appeared in the ANSI/C and 7.40 releases of the debugger. In summary, the following features have been added:

- Limited support for the ANSI/C "long double" data-type (copy and display; no support for full expressions).
- Full NLS support for the display and modification of the "wchar_t" data-type; mapping is determined by LC_CTYPE locale.
- Support for new character escape-sequences "\?" and "\a".
- Support for new "unlimited hex" character escape-sequence "\xFFFF".
- Support for new "wide-character" prefix "L".
- Support for new floating-point suffixes "F", "f", "L", and "l".
- Support for new unsigned suffixes "u" and "U".

3-6 Part 2: Release-specific Topics



C++ and Debugger

The 8.0 debugger provides full support for C++ types and operators, and provides several new commands for debugging C++ objects. It is fully compatible with the 2.0 release of the HP C++ product, and most of the following features appeared in the debugger provided with that product. In summary, the following features have been added:

- Support for C++ types including classes, references, pointers to class members, and anonymous unions.
- Support for the :: operator.
- Support for new class and declaration scopes.
- Support for simple C++ expressions which do not rely upon elaborate type conversion mechanisms such as automatic constructor invocation.
- Support for syntax for referencing variables which is identical to C++ and extended to handle existing debugger functionality.
- Support for overloaded operators specified as operator <op>.
- Support for command line procedure calls of C++ functions.
- Support for class type casts.
- Where ambiguity arises in C++ debugging due to overloaded function names, provided the user with a friendly interface to interactively disambiguate them.
- Extended existing print mechanisms and syntaxes to support C++ and added the "k", "K", and "T" print formats to allow the user to print additional class information.
- Provided additional listing capabilities for listing class member functions and overloaded functions.
- Extended existing view commands to handle member functions.
- Support for breakpoints on member functions.
- Added the "bpo" command to set breakpoints on overloaded functions.
- Added the "bpc" command to set breakpoints on class member functions.

- Extended the “bp” command to set breakpoints on the member functions of a given class.
- Support for instance breakpoints.

ther Enhancements

- Support for input and display of binary integers.
- Support for save/restore of the current macro, breakpoint, and assertion state.
- Included beginning/ending source line-numbers in the output from the “list procedures” command.
- Enhanced goto command accepts +/- offsets from the current location, in source or disassembly mode.
- Added a “quiet” option (“pq”) to the print command. This option enables silent assignments in assertions and breakpoint command lists.
- Revised the on-line help facility to provide help on a command and topic basis.

Compatibility

The Series 300/400 debuggers are not compatible with 7.0 compilers or the (Series 300/400) ANSI/C compiler. Programs compiled with these compilers must be recompiled before they can be debugged with the 8.0 debugger. This incompatibility is due to the convergence with the PA-RISC implementation of the format used for symbolic debug information. Object files or libraries previously compiled with -g using these compilers may be linked to a program being debugged on 8.0, but the symbolic information in these modules will not be used.

The Series 300/400 debuggers are fully compatible with the compilers in the 7.40 Languages release, as well as the C++/2.0 compiler.

The Series 300/400 debuggers are fully compatible with the C++/2.0 product, and may be used in place of the debuggers provided with it (cdb++, xdb++).

3-8 Part 2: Release-specific Topics

Impact on Performance

A separate program (“pxdb”) now handles the preprocessing of debug information, as on the PA-RISC implementation. By default, this preprocessor is automatically invoked by the linker as part of the final link phase of a compilation where the “-g” option was specified. As a result of this change, the start-up time and overall performance of the debugger, especially on large applications, is much improved.

Other improvements include enhancements to speed up the display updates in the source and disassembly windows, and static table searches (linker symbol tables).

User Limitations

As noted earlier, the debugger can operate in the presence of shared-libraries, but breakpoints cannot be set in them, and they cannot be single-stepped into. This is due to their very nature: they are shared, so they cannot be modified.

Symbolic Debugger, Series 600/800

Several new features are available with the 8.0 release of the HP Symbolic Debugger. The user is encouraged to examine the new manual:

HP Symbolic Debugger User's Guide (P/N B1699-90601)

Shared libraries

The most important new feature is the ability to debug programs that have been linked with shared-libraries. Although symbolic debugging of linked shared-libraries is not supported, the user has the ability to view shared-library code in disassembly mode, set or examine data associated with the library, or call shared-library functions from the command line. The linker symbol table for the library is automatically loaded by the debugger, so all address within the library may be referenced symbolically via a special syntax using “@” (e.g. “_printf@libc”).

Breakpoints may not be set with shared-library functions, and the user cannot single-step into one; instead, the debugger steps “over” shared-library calls as if they were system calls.

C++ and Debugger

The 8.0 debugger provides full support for C++ types and operators, and provides several new commands for debugging C++ objects. It is fully compatible with the 2.0 release of the HP C++ product, and most of the following features appeared in the debugger provided with that product. In summary, the following features have been added:

- Support for C++ types including classes, references, pointers to class members, and anonymous unions.
- Support for the :: operator.
- Support for new class and declaration scopes.
- Support for simple C++ expressions which do not rely upon elaborate type conversion mechanisms such as automatic constructor invocation.
- Support for syntax for referencing variables which is identical to C++ and extended to handle existing debugger functionality.
- Support for overloaded operators specified as operator <op>.
- Support for command line procedure calls of C++ functions.
- Support for class type casts.
- Where ambiguity arises in C++ debugging due to overloaded function names, provided the user with a friendly interface to interactively disambiguate them.
- Extended existing print mechanisms and syntaxes to support C++ and added the “k”, “K”, and “T” print formats to allow the user to print additional class information.
- Provided additional listing capabilities for listing class member functions and overloaded functions.
- Extended existing view commands to handle member functions.
- Support for breakpoints on member functions.
- Added the “bpo” command to set breakpoints on overloaded functions.
- Added the “bpc” command to set breakpoints on class member functions.

3-10 Part 2: Release-specific Topics

- Extended the “bp” command to set breakpoints on the member functions of a given class.
- Support for instance breakpoints.

PA-RISC Float Register display

The floating-point (FP) register display for PA-RISC implementations has been improved to show both hexadecimal and single or double-precision values, as it is on the Series 300/400 debuggers. The “tf” (toggle float) command toggles the precision displayed. Additionally, the window is automatically scrolled to show FP registers as they change, and a register map highlights the name of all changed registers. A status line shows all fields and flags of the FP coprocessor status-word.

All FP registers are now accessible through pre-defined special variables. The left (high-order) or right (low-order) half of each FP register can be referenced as a single-precision quantity with “\$f??l” or “\$f??r” (“??” being a 1 or 2 digit quantity from 0 to 15). The entire 2-word register can be referenced as a double-precision quantity with “\$f??”.

Other Enhancements

- Support for input and display of binary integers.
- Support for save/restore of the current macro, breakpoint, and assertion state.
- Included beginning/ending source line-numbers in the output from the “list procedures” command.
- Enhanced goto command accepts +/- offsets from the current location, in source or disassembly mode.
- Added a “quiet” option (pq) to the print command. This option enables silent assignments in assertions and breakpoint command lists.
- Revised the on-line help facility to provide help on a command and topic basis.

Compatibility

The PA-RISC debugger is compatible with the 7.0 compilers. Programs linked (but not already debugged) on 7.0 systems can, in general, be debugged on 8.0. This also applies to programs linked against 7.0 libraries. Programs that have already been debugged on 7.0 will not be debuggable on 8.0 due to changes in the debug-info preprocessor (“pxdb”). In these cases the debugger will issue a message indicating the user should re-link his application.

In certain cases where the 7.0 compilers produced incorrect symbolic debug information (with the “-g” compile option), the preprocessor will detect this and issue an internal error. In such cases, the user should re-compile his application to insure that correct debug information is generated and available to the new debugger.

The PA-RISC “xdb” is fully compatible with the C++/2.0 product, and may be used in place of xdb++.

Impact on Performance

The symbolic debug info preprocessor (“pxdb”) is now automatically invoked by the linker as part of the final link phase of a compilation where the “-g” option was specified. This may significantly reduce the start-up time of the debugger, particularly for large applications.

Other improvements include enhancements to speed up the display updates in the source and disassembly windows, and static table searches (linker symbol tables, unwind tables).

Support

The file “/usr/lib/xd bend.o” is being replaced by the file “/usr/lib/end.o” (which is used on the Series 300/400). For the 8.0 release, both paths refer to the same file (they are linked together).

The 7.0 compilers automatically included the file “/usr/lib/xd bend.o” in the linker command line when “-g” was specified. Users who explicitly invoke “ld(1)” for the final link phase of a debuggable program should also be including this file. They are encouraged to replace such references with references to “/usr/lib/end.o”.

User Limitations

As noted earlier, the debugger can operate in the presence of shared-libraries, but breakpoints cannot be set in them, and they cannot be single-stepped into. This is due to their very nature: they are shared, so they cannot be modified.

_Documentation

Several changes were made in the HP-UX documentation set between the 7.0 and 8.0 releases.

- In many cases, you will note significant changes in the part numbers and the part-numbering scheme.
- The set for 8.0 has some new documents, for example, an error message catalog and a manual for cluster management.
- The concepts and tutorial manuals were reorganized and restructured, becoming a series of HP-UX user's guides.
- There is now only one beginner's guide.
- Many spines and covers were redesigned to facility accessibility.
- The networking reference pages were merged into the HP-UX Reference manuals.
- There is an online reference manpage that lists manuals/editions for this release.
- There is more online information: a CD-ROM product, online help for "TWM" and "keysh", and more help screen for SAM.

The following tables list document for the 8.0 release and, in some cases, they show corresponding documents for the 7.0 release. As you read, note any differences among documents for particular series.

HP 9000 Computers Series 600/800 Documentation

System Documents for Series 600/800

8.0 Title/Part Number	7.0 Title/Part Number
-----	-----
A Beginner's Guide to HP-UX B1862-90000	* A Beginner's Guide to HP-UX CO#: 98594-90006
	* A Beginner's Guide to Shells CO#: 98594-90008
	* A Beginner's Guide to Text Editing CO#: 98594-90010
	* Introducing the UNIX System V Series 300 only at 7.0 CO#: 98597-90620 (McGraw-Hill Book Company)
	* Quick Reference for vi CO#: 98597-90000
Finding Information B2437-90001	* Finding HP-UX Information CO#: 92453-90042
Installing and Updating HP-UX B2437-90003	* Installing HP-UX CO#: 92453-90035
Master Index B2437-90004	* HP-UX Master Index CO#: 92453-90041
Error Message Catalog B1862-90004	* New manual at 8.0 Some information from: * Troubleshooting HP-UX Systems CO#: 92453-90026
How HP-UX Works Concepts for Sys Admin B1862-90005	* HP-UX System Administration Concepts Manual CO#: 98594-90062
Managing Clusters of HP 9000 Computers Sharing the HP-UX File System B1862-90006	* New manual

8.0 Title/Part Number -----	7.0 Title/Part Number -----
Installing Peripherals B2437-90005 (2 Vols)	* HP-UX Installing Peripherals CO#: 92453-90032
System Administration Tasks B2437-90006	* HP-UX System Administration Tasks CO#: 92453-90038
HP-UX System Security B1862-90009	* HP-UX System Security CO#: 92453-90029
Solving HP-UX Problems B1862-90010	* Troubleshooting HP-UX Systems CO#: 92453-90026
Remote Access: User's Guide B1862-90011	* Concepts and Tutorials UUCP CO#: 97089-90053 * Concepts and Tutorials Shells and Miscellaneous Tools CO#: 97089-90062
Mail Systems: User's Guide B1862-90012	* Concepts and Tutorials Shells and Miscellaneous Tools CO#: 97089-90062
Terminal Control: User's Guide B1862-90013	* Concepts and Tutorials Device I/O and User Interfacing CO#: 97089-90057 * Concepts and Tutorials Facilities for 200/300/500 - Series 300 only at 7.0 CO#: 97089-90081 * Using the HP 9000 ITE - S800 only for 7.0, this book was obsoleted 01/30/90 CO#: 92453-90011 * Term0 Reference Manual Series 300 only at 7.0 CO#: 82865-90002

3-16 Part 2: Release-specific Topics

8.0 Title/Part Number	7.0 Title/Part Number
-----	-----
Text Formatting: User's Guide B1862-90014	* Concepts and Tutorials Text Formatters CD#: 97089-90032
Number Processing: User's Guide B1862-90015	* Concepts and Tutorials Shells and Miscellaneous Tools CD#: 97089-90062
Text Processing: User's Guide B1862-90016	* Concepts and Tutorials Text Editors and Processors CD#: 97089-90022
Shells: User's Guide B1862-90016	* Concepts and Tutorials Shells and Miscellaneous Tools CD#: 97089-90062 * A Beginners Guide to Shells CD#: 98594-90008 * Application Execution Environment User's Guide - Series 300 only at 7.0 CD#: 98515-90004
HP-UI Reference B1864-90000	* HP-UI Reference CD#: 09000-90013
POSIX Conformance Document B1864-90001	* POSIX Conformance Document CD#: 98794-90009
Device I/O: User's Guide B1864-90002	* Concepts and Tutorials Device I/O and User Interfacing CD#: 97089-90057 * Concepts and Tutorials Facilities for 200/300/500 - S300 only at 7.0 CD#: 97089-90081

8.0 Title/Part Number -----	7.0 Title/Part Number -----
Native Language Support: User's Guide B1864-90003	* Concepts and Tutorials Native Language Support CD#: 97089-90058
Terminal Session Manager: User's Guide B2351-9000	* New manual at 8.0
The Ultimate Guide to the vi and ex Text Editors 97005-90015 (Benjamin Cummings Pub. Company, Inc.	* Concepts and Tutorials Text Editors and Processors CD#: 97089-90022 * A Beginner's Guide to Text Editing CD#: 98594-90010
92625A Disk Mirrors Data/Pair Managing Disk Mirrors Using DataPair/800 CD#: 92453-90023	* Managing Disk Mirrors Using DataPair/800 same CD#: 92453-90023
92668A SwitchOver/UX Product Managing SwitchOver/UX 92668-90001	* New product and document
HP-UX Symbolic Debugger User's Guide B1864-90005	Not available at 7.0
HP-UX Portability Guide B1864-90006	HP-UX Portability Guide 97033-90046
Programming on HP-UX B1864-90007	HP-UX Concepts and Tutorials Programming Environment 97089-90042

3-18 Part 2: Release-specific Topics

Graphics Documents for Series 600/800

8.x PN/Title		7.x PN/Title	
98592-90025	Starbase Device Drivers Library	98592-90019	Starbase Device Drivers Library
98592-90047	A Beginner's Guide to Using Starbase	98592-90046	A Beginner's Guide to Using Starbase
98592-90067	Starbase Reference	98592-90066	Starbase Reference
98592-90080	Starbase Graphics Techniques	98592-90079	Starbase Graphics Techniques
98592-90093	FA/FM Programmer's Manual	98592-90092	FA/FM Programmer's Manual
98672-90602	GKS Func. Desc. ANSI Standard	Same	
98885-90002	Starbase Radiosity Ray Tracing Prog. Mn	98885-90001	Starbase Radiosity and Ray Tracing Prog. Mn
B2361-90000	Starbase Display List Programmer's Mn	98674-90003	Starbase Display List Programmer's Mn
B2362-90000	HP-GKS Device Drivers Library	98672-90014	HP-GKS Device Drivers Library
B2362-90001	GKS Users Guide	98672-90001	HP-GKS Users Guide
97084-90011	DGL Programmer's Manual Series 800 only.	97084-90010	DGL Programmer's Manual Series 300/800.
97084-90029	DGL/AGP Device Drivers Manual Series 800 only.	97084-90028	DGL/AGP Device Drivers Manual Series 300/800.
97085-90013	AGP Programmer's Manual Series 800 only.	97085-90012	AGP Programmer's Manual Series 300/800.

Other HP-UX Learning Products for Series 600/800

8.0 Title/Part Number

Assembler

92432-90001 Assembly Language Reference

C

92453-90024 HP C/HP-UX Reference Manual

FORTRAN

92430-90004 HP FORTRAN 77/HP-UX Prog. Guide

92430-90005 HP FORTRAN 77/HP-UX Reference Manual

Pascal

92431-90005 HP Pascal/HP-UX Reference Manual

92431-90006 HP Pascal/HP-UX Programmer's Guide

92431-90007 HP Pascal/HP-UX Quick Ref. Guide

HP 9000 Computers Series 300/400 Documentation

System Documents

8.0 Title/Part Number

A Beginner's Guide to HP-UX
B1862-90000

Finding Information
B1862-90001

Installing and Updating HP-UX
B1862-90002

Master Index
B1862-90003

Error Message Catalog
B1862-90004

7.0 Title/Part Number

- * A Beginner's Guide to HP-UX
CD#: 98594-90006
- * A Beginner's Guide to Shells
CD#: 98594-90008
- * A Beginner's Guide to Text Editing
CD#: 98594-90010
- * Introducing the UNIX System V
Series 300 only at 7.0
CD#: 98597-90620
(McGraw-Hill Book Company)
- * Quick Reference for vi
CD#: 98597-90000

* Finding HP-UX Information
CD#: 98594-90081

* Installing HP-UX
CD#: 98594-90013

* HP-UX Master Index
CD#: 98594-90090

- * New manual at 8.0
Some information from:
- * Troubleshooting HP-UX Systems
CD#: 92453-90026

8.0 Title/Part Number	7.0 Title/Part Number
-----	-----
How HP-UX Works Concepts for the Sys Admin B1862-90005	* HP-UX System Administration Concepts Manual CO#: 98594-90062
Managing Clusters of HP 9000 Computers Sharing the HP-UX File System B1862-90006	* New manual at 8.0
Installing Peripherals B1862-90007 (2 Vols)	* HP-UX Installing Peripherals CO#: 97005-90002
System Administration Tasks B1862-90008	* HP-UX System Administration Tasks CO#: 98594-90061
HP-UX System Security B1862-90009	* HP-UX System Security CO#: 92453-90029
Solving HP-UX Problems B1862-90010	* Troubleshooting HP-UX Systems CO#: 92453-90026
Remote Access: User's Guide B1862-90011	* Concepts and Tutorials UUCP CO#: 97089-90053 * Concepts and Tutorials Shells and Miscellaneous Tools CO#: 97089-90062
Mail Systems: User's Guide B1862-90012	* Concepts and Tutorials Shells and Miscellaneous Tools CO#: 97089-90062

3-22 Part 2: Release-specific Topics

8.0 Title/Part Number	7.0 Title/Part Number
Terminal Control: User's Guide B1862-90013	* Concepts and Tutorials Facilities for Series 200/300/500 CD#: 97089-90081 * Concepts and Tutorials Device I/O and User Interfacing CD#: 97089-90057 * Term0 Reference Manual Series 300 only at 7.0 CD#: 82865-90002 * Using the HP 9000 ITE - Series 800 only at 7.0 book was obsoleted 01/30/90 CD#: 92453-90011
Text Formatting: User's Guide B1862-90014	* Concepts and Tutorials Text Formatters CD#: 97089-90032
Number Processing: User's Guide B1862-90015	* Concepts and Tutorials Shells and Miscellaneous Tools CD#: 97089-90062
Text Processing: User's Guide B1862-90016	* Concepts and Tutorials Text Editors and Processors CD#: 97089-90022
Shells: User's Guide B1862-90017	* Concepts and Tutorials Shells and Miscellaneous Tools CD#: 97089-90062 * A Beginner's Guide to Shells CD#: 98594-90008 * Application Execution Environment User's Guide CD#: 98515-90004

8.0 Title/Part Number	7.0 Title/Part Number
-----	-----
HP-UX Reference B1864-90000	* HP-UX Reference CO#: 09000-90013
POSIX Conformance Document B1864-90001	* POSIX Conformance Document CO#: 98794-90009
Device I/O: User's Guide B1864-90002	* Concepts and Tutorials Device I/O and User Interfacing CO#: 97089-90057 * Concepts and Tutorials Facilities for Series 200/300/500 CO#: 97089-90081
Native Language Support: User's Guide B1864-90003	* Concepts and Tutorials Native Language Support CO#: 97089-90058
The Ultimate Guide to the vi and ex Test Editors 97005-90015 (Benjamin Cummings Pub. Company, Inc.)	* Concepts and Tutorials Text Editors and Processors CO#: 97089-90022 * A Beginner's Guide to Text Editing CO#: 98594-90010
B2351A Product Terminal Session Manager: User's Guide B2351-90000	* New Product at 8.0

3-24 Part 2: Release-specific Topics

Graphics Documents for Series 300/400

8.x PN/Title	7.x PN/Title
98592-90025 Starbase Device Drivers Library	98592-90019 Starbase Device Drivers Library
98592-90047 A Beginner's Guide to Using Starbase	98592-90046 A Beginner's Guide to Using Starbase
98592-90067 Starbase Reference	98592-90066 Starbase Reference
98592-90080 Starbase Graphics Techniques	98592-90079 Starbase Graphics Techniques
98592-90093 FA/FM Programmer's Manual	98592-90092 FA/FM Programmer's Manual
98672-90602 GKS Func. Desc. ANSI Standard	Same
98885-90002 Starbase Radiosity Ray Tracing Prog. Mn	98885-90001 Starbase Radiosity and Ray Tracing Prog. Mn
B1759-90000 HP-PHIGS Graphics Techniques Mn.	B1759-90002 HP-PHIGS Graphics Techniques Mn.
B1759-90003 A Beginner's Guide To Using HP-PHIGS	B1759-90002 A Beginner's Guide To Using HP-PHIGS
B1759-90005 HP-PHIGS FORTRAN Binding Reference	B1759-90001 HP-PHIGS FORTRAN Binding Reference
B1759-90006 HP-PHIGS Workstation Characteristics and Implement. Manual	B1759-90002 HP-PHIGS Workstation Characteristics and Implement. Manual
B1760-90000 Introducing the Personal Visualizer	Same
B1760-90001 Personal 3D Edit User's Guide	Same

8.x PW/Title		7.x PW/Title	
B1760-90002	Personal Visualizer User's Guide (w/update)	B1760-90002	Personal Visualizer User's Guide
B1760-90003	Personal Translator: IGES User's Guide	Same	
B1760-90004	Personal Translator: I-DEAS Solid Modeling User's Guide	Same	
B1760-90006	Personal Visualizer HP Installation and Configuration Guide	B1760-90005	Personal Visualizer HP Installation and Configuration Guide
None	In Starbase Device Drivers Library	B1760-90605	Starbase to Personal Visualizer Archive Device Driver
B2360-90000	HP Windows/9000 Documentation Series 300 only.	97069-90003	HP Windows/9000 Documentation
B2360-90001	HP Windows/9000 Reference Series 300 only.	97069-90023	HP Windows/9000 Reference
B2361-90000	Starbase Display List Programmer's Mn	98674-90003	Starbase Display List Programmer's Mn
B2362-90000	HP-GKS Device Drivers Library	98672-90014	HP-GKS Device Drivers Library
B2362-90001	GKS Users Guide	98672-90001	HP-GKS Users Guide
97084-90011	DGL Programmer's Manual Series 800 only.	97084-90010	DGL Programmer's Manual Series 300/800.
97084-90029	DGL/AGP Device Drivers Manual Series 800 only.	97084-90028	DGL/AGP Device Drivers Manual Series 300/800.
97085-90013	AGP Programmer's Manual Series 800 only.	97085-90012	AGP Programmer's Manual Series 300/800.

3-26 Part 2: Release-specific Topics

Language Documents for Series 300/400

Release 7.x -----	Release 8.0 -----
FORTRAN/9000 Reference B2372-90000	FORTRAN/9000 Reference 98598-90041
Pascal Language Reference B2373-90000	Pascal Language Reference 98615-90053
C Programmer's Guide B1864-90008	C Programmer's Guide 98794-90030
C Programming Tools B1864-90009	Not available at 7.0
C: A Reference Manual B1700-90002	C: A Reference Manual B1700-90000
HP-UX Symbolic Debugger User's Guide B1864-90005	Not Available at 7.0
HP-UX Assembler and Tools B1864-90004	HP-UX Assembler Tools 98597-90020
HP-UX Portability Guide B1864-90006	HP-UX Portability Guide 97033-90046
Programming on HP-UX B1864-90007	HP-UX Concepts and Tutorials Programming Environment 97089-90042

Other Learning Products for Series 300/400

8.0 Title/Part Number

Assembler

92432-90001 Assembly Language Reference

C

92453-90024 HP C/HP-UX Reference Manual

FORTRAN

92430-90004 HP FORTRAN 77/HP-UX Prog. Guide

92430-90005 HP FORTRAN 77/HP-UX Reference Manual

Pascal

92431-90005 HP Pascal/HP-UX Reference Manual

92431-90006 HP Pascal/HP-UX Programmer's Guide

92431-90007 HP Pascal/HP-UX Quick Ref. Guide

Networking Documents for HP 9000 Systems

Title -----	HP-UX 8.0 Customer Order No.	HP-UX 7.0 Customer Order No.
	-----	-----
Using Network Services	B1012-90009	B1012-90000
Inst. and Admin. Network Services	B1012-90010	B1012-90001
Networking Overview	B1012-90012	B1012-90003
NS Network Reference Pages	n/a	B1012-90002
Using NFS Services	B1013-90008	B1013-90000
Inst. and Admin. NFS Services	B1013-90009	B1013-90001
Programming and Protocols for NFS Services	B1013-90010	B1013-90002
NFS Network Reference Pages	n/a	B1013-90003
Using ARPA Services	B1014-90006	B1014-90000
Inst. and Admin. ARPA Services	B1014-90007	B1014-90001
ARPA Network Reference Pages	n/a	B1014-90002

8.0 Title/Part Number	7.0 Title/Part Number	
HP OSI Express 802.4 Hardware Inst. and Config. Guide	28604-90001	28604-90001
Inst. and Admin. HP OSI Express MAP 3.0 Link	32122-90001	32122-90001
Admin. Ref. Manual for HP OSI Express MAP 3.0 Link	32122-90003	32122-90003
Troubleshooting HP OSI Express MAP 3.0 Link	32122-90002	32122-90002
Read Me First	36969-60500	n/a
HP MAP 3.0 MMS/800 Ref. Manual	n/a	B1002-90000
HP MAP 3.0 MMS/800 Progrm. Guide	n/a	B1002-90010
HP MAP 3.0 FTAM/800 Ref. Manual	n/a	B1032-91000
HP MAP 3.0 FTAM/800 Progrm. Guide	n/a	B1032-91010
HP MAP 3.0 FTAM/800 User's Guide	n/a	B1032-91020
HP MMS/9000 Ref. Manual	32019-60500	B1002-90000
HP MMS/9000 Progrm. Guide	32019-60510	B1002-90010
HP FTAM/9000 Ref. Manual	B1033-60500	B1032-90000 and B1004-90000
HP FTAM/9000 Progrm. Guide	B1033-60510	B1032-90010 and B1004-90000
HP FTAM/9000 User's Guide	B1033-60520	B1032-90020
OSI Config. and Trouble Guide	32069-60001	32069-60001
Inst. and Admin. OSI Transport Services	32069-60003	32069-60003
XTI Programmer's Guide	32069-60002	32069-60002
Session Level Progrm. Guide	32069-60004	n/a
Inst. and Admin. X.400/9000	32034-90001	32034-90001
HP X.400/HP Desk Admin. Guide	32055-90001	32055-90001
Using HP DeskMgr Connec. X.400	32055-90002	32055-90002

_Drivers

Autochanger driver (C1700A Optical)

Summary of Change

The major change is to expand support for the C1700A optical autochanger to the series 800 system. Support will continue for the series 300 but will include some enhancements made to the driver. The major enhancement for the series 300 is that an “fsck” of every mounted disk is not required after a power failure or a system crash.

The name of the driver has been changed on the 300 from “ac” to “autoch”. The master file has been changed to reflect that fact but any old dfiles that contain a reference to “ac” should not be used.

Configuration

Since the C1700A is a SCSI only peripheral, the autochanger driver will only be supported on a system that has a SCSI interface. The driver will be supported on both NIO and CIO cards on the 800 and on the 98658A and the 98265A SCSI cards on the 300.

Features

The user treats the autochanger as 64 mountable disks (32 cartridges, with two sided media). All 32 cartridges can be mounted into the file system and can be accessed transparently. No special program needs to be executed nor any user intervention is needed to load the proper cartridge into a drive. Implied in transparent access is the ability to access surfaces without worrying about synchronizing several users. Just as several users may access the file systems on disks so too can they access file on different cartridges. The autochanger driver takes care of this synchronization making sure that the proper media surface is in the drive.

Existing application programs do not have to change to access files on the autochanger.

Disk sections are supported on the cartridges on the series 800 only.

Impact

A new entry has been made in the `bdevsw[]` table to facilitate the marking of the surface clean before it is removed from the drive. This prevents the need to “fsck” the surface after a power failure. This should be transparent to all existing drivers including 3rd party drivers. The “conf.c” program created by “config” or “uxgen” will reflect this new entry point.

There are two situations that can occur which will prevent the autochanger driver from marking the surface clean. These are if there are:

1. any pipes configured on the surface, and
2. if there are any unlinked-open files on the surface.

Application programs should be careful not to create these situations on a surface. They will operate properly but if a power failure or system crash occurs, that surface must be fscked. The situations arise primarily for use in temporary space allocation which for performance reasons should reside on online disk.

Performance data

Because of the mechanical nature of the device, if the file requested is on a cartridge which is not currently in a drive, a 15 second delay will occur while the cartridge is inserted. Once the cartridge is inserted, the user performance will be identical to that of the C1701A, stand-alone optical disk. Users need to be aware that when accessing files on the autochanger, some delay will occur. Treating this device as a straight winchester replacement will lead to disappointing performance.

Several performance enhancements have been made over the 7.0 driver. Since each surface has all its buffers flushed to the disk and the superblock marked clean before it is removed from the drive and put back into a slot, the syncer will not cause excessive swapping when several cartridges are mounted read-write.

Another improvement is that after the last close of a surface, the cartridge is left in the drive for a short period of time before it is put back into its slot. If the cartridge is accessed again before this time expires, an additional swap of the cartridge is not required. This will happen when a user does a “mediainit”, “newfs”, and “mount” of the cartridge.

Size requirement

The kernel data structures needed in the driver are dynamically allocated and therefore the memory requirements are dependent upon the system load.

The size of the driver is about 5000 lines of C source with the resulting object files of about 47k on the series 300 and about 72k on the series 800.

NOTE: When configuring a system with an autochanger, the number of system buffers and inodes should be increased. Each mounted file system takes a buffer and 4k bytes of memory to hold the superblock in core. If several autochangers are connected, with each surface split into sections (on an 800), the number of buffers and inodes required becomes large.

centronics (HP-PB)**Summary of Change**

This is a new driver. The new driver is named "lpr2".

Configuration

The centronics printer driver "lpr2" is added to the system by adding an include statement (for example, "include lpr2;") in the gen file.

Features

This driver supports all the standard printer features. Both cooked and raw modes are supported plus io control.

The "slp" command may be used to perform io control on the printer.

Impact

There is no difference in functionality. In some applications an increase in through-put may be seen.

CS80 device driver (disc1)

Summary of Change

By using CS80 device driver (“disc1”) for 8.0, both CS80 devices and magnetic tape devices can be used on the same HP-IB interface card for CIO machines. For NIO machines, no change is included. (7.0 system can already support both disks and tapes on the same bus.) Memory core dump feature of CS80 driver has been removed. Instead, IODC is used to do memory core dump.

Configuration

Simply type “include disc1” in your “uxgen” file for s800 CIO machines. No change is required for NIO machines except memory core dump feature has been removed.

Features

By using CS80 device driver for 8.0 (disc1), both CS80 devices and magnetic tape devices can be used on the same HP-IB interface card for CIO machines. Also, memory core dump uses IODC path, which is more robust than a disk driver path.

Impact

The uxgen file has to have a statement “include disc1” instead of “include disc0”. Disc0 driver is not supported for 8.0 system. Also make sure that disk driver entries for “/etc/master” file have “no dump” in their function fields. This forces memory core dump to use IODC code.

Size requirement

CS80 device driver will require 49K bytes of memory compared to 33K bytes for 7.0. This is because the low driver manager code (hpib0) has to be included in the system. There is no size change for NIO machines.

SCSI Supported on Series 800

Content

SCSI is supported on Series 800 as of HP-UX release 8.0. The newly added Series 800 SCSI implementation includes SCSI interfaces that are available for both the NIO and the CIO backplanes. Many key new Hewlett-Packard SCSI mass storage peripherals will be supported as well as several existing Hewlett-Packard SCSI peripherals.

The three device classes that the Series 800 SCSI implementation supports are direct-access (disks), sequential-access (tapes), and autochanger.

Compatibility

The Series 800 SCSI implementation is targeted toward HP Common SCSI (an HP interpretation of the standard SCSI-2 specification). In addition, the Series 800 implementation is compatible with ANSI SCSI-1 and SCSI-2 specifications.

The new SCSI interface is usable in systems which may already include HP-IB or HP-FL interfaces.

System features for supported devices include:

- Boot from both Sequential (DDS) and Direct-Access (disk) Devices
- Autoconfiguration
- System Memory Dump to Direct-Access Devices
- Sherlock Diagnostics

The following is a peripheral support matrix:

Model	Device Type	Boot Device	System Disk
2212A	Direct-Access	Yes	Yes
2213A	Direct-Access	Yes	Yes
7957S	Direct-Access	Yes	Yes
7958S	Direct-Access	Yes	Yes
7959S	Direct-Access	Yes	Yes
97536S	Direct-Access	Yes	Yes
97544S	Direct-Access	Yes	Yes
97548S	Direct-Access	Yes	Yes
35450	Sequential	Yes	N/A
C1512A	Sequential	Yes	N/A

3-34 Part 2: Release-specific Topics



7980S	Sequential	Yes	N/A
C1701A	Magneto Optical	Yes	No
C1700A	MO Disk Library	No	No

Many third party devices would adversely impact many of the system features previously mentioned. Use of third-party peripherals is at the user's risk, and is unsupported by HP.

Dependencies

No new commands were implemented (only command enhancements) to support SCSI. A few new autoconfiguration commands (which support interface cards used in CIO or NIO backplanes) will appear in 8.0 and are required.

Commands enhanced for SCSI support include MEDIAINIT, INSF, LSSF, MKSF.

Miscellaneous files added/enhanced for SCSI support include:

```
/usr/include/sys.scsi.h
/etc/master
diskio.h
/etc/disktab
```

The Direct-Access class of SCSI device drivers will support the standard kernel entry points: open(), close(), read(), write(), ioctl().

Block-special and character-special (raw) device access are supported as in past releases (as described in the disk(7) man-page).

The Sequential-Access class of SCSI device drivers will support the standard kernel entry points: open(), close(), read(), write(), ioctl().

User Limitations

Certain characteristics of SCSI (an industry-standard bus) preclude the removal/connection of SCSI cables from running systems.

_File Systems and Files

fs_async and Synchronous Disk Writes

Features

In the previous release of HP-UX, many file system data structures were written to disk synchronously. While this strategy helped insure file system integrity in the event of a crash, it was costly in terms of file system performance. Performance tests indicated that system performance could increase significantly when disk writes occurred asynchronously.

A new feature available on both the series 300 and 800 allows the user to specify whether these disk writes should be performed synchronously or asynchronously. If asynchronous writes are selected, UNIX file system semantics for NFS and diskless clustered environments are preserved. If a file is opened with the "O_SYNC" option and asynchronous writes are selected, the file will continue to be written synchronously.

Configuration

The "fs_async" parameter in the S800 gen file and the S300 dfile enables and disables the feature. A zero value specifies that the writes should be performed synchronously and a non-zero value specifies that the writes should be performed asynchronously. If this parameter is not specified, then writes are performed synchronously.

Impact

If this feature is enabled, system performance should increase for most applications. However, in the event of a system crash, file system data structures may be left in a more inconsistent state than if the feature was not enabled. Normally, recovery from a file system crash is performed automatically by "fsck" in the reboot process and does not require any system administrator intervention. However, use of this feature may require system administrator intervention in the event of a crash.

Performance data

Performance testing using a program development benchmark on the series 300 and 800 revealed that some performance gain was realized in all test cases when disk writes were done asynchronously. On the series 800, performance improvements of 6 to 26 per cent were obtained with between 3 and 30 users. The series 300 tests achieved performance improvements of 6 to 25 per cent with between 1 and 12 users.

Disk Quotas and Commands

Added Functionality

Disk Quotas provide a mechanism for controlling, and quickly reporting on, user utilization of disk space. Soft (quota) and Hard (limit) bounds may be set by the super-user on number of files and number of blocks individual users may utilize on each file system. Quotas are enforced after a file system dependent timeout (which defaults to 7 days), while limits are enforced constantly. Enforcement means that user attempts to claim more resource are denied.

An example is a user with a disk-block quota of 10000 and limit of 20000 on the mounted /users file file system. When the user's disk usage exceeds 10000 1024 byte blocks, they receive an immediate warning and also begin receiving warnings at subsequent logins. If the user ignores warnings to reduce disk usage as instructed, after 7 days the user's attempts to claim more blocks are denied by the kernel. Programs, including login shells, then behave as if the file system were out of space for that user. For up to 7 days the user may continue to exceed the quota, but not the limit. If, at any time, the user attempts an operation which would exceed the 20000 block limit, that operation fails in the same way. The user may return to normal operation from either state by reducing file system utilization below the 10000 block quota.

HP Enhancements

HP-UX disk quotas are based on Sun Microsystems disk quotas, which were in turn leveraged from BSD4.3. The HP system programmatic interface is identical to Sun. HP has enhanced disk quotas to improve robustness by adding the ability to enable quotas at mount time rather than having to wait until after the mount. This avoids the window where the file system is mounted but not under quota control. File systems properly configured will have quotas

enabled by virtue of the `/etc/checklist` or `mount` command-line options. As a result of this feature, the `quotaon` and `quotaoff` commands are not generally required and their use is discouraged to prevent having to use the `quotacheck` command at boot time. `Quotacheck` need only be run when establishing new quotas for a new file system user.

Another enhancement is the addition of the “`fs_qclean`” flag in the file system super block which indicates the validity of the quotas file information. This flag is maintained by the “`quotactl(2)`”, “`vfsmount(2)`”, “`mount(2)`”, and “`umount(2)`” system calls and the commands that use them. The “`/etc/rc`” script has been enhanced to support disk quotas and to `quotacheck` when necessary.

Implications with respect to other features

Systems employing disk quotas should protect against the use of the `chown` command and system call by users attempting to hide their disk usage. This may be done with the “`setprivgrp(1M)`” command as illustrated in “`/etc/newconfig/rc`”.

Performance Impact as a result of Disk Quotas

There are two levels of Disk Quota configuration and overhead:

- Configuration on the file system
- Configuration for an individual user

The effects on performance are cumulative.

General Overhead

File system allocation/deallocation operations result in test to see if the affected file system has quotas enabled.

Configuration on a file system

If disk quotas are enabled on the affected file system, the system must check to see if the user has quotas there. If not, disk accounting operations are skipped.

Effect of quotas for individual users

If the user is found to have quotas set, each operation described above is accompanied by compare and update operations, plus additional actions if the operation causes the quota to be exceeded. No quota accounting is performed for users who have no quotas set.

System Boot Overhead

When the system is booted after a crash, statically stored disk quota information must be collected with “quotacheck(1M)” for each file system on which quotas are enabled. This requires scanning the entire file system and recording usage. This typically takes about 1/6 as long as an “fsck” operation on the same file system.

If the system is shut down cleanly (i.e. with “reboot” or “shutdown”), the quota statistics are automatically saved and marked valid so that no quota check is required. The root (“/”) file system is an exception in that it must always be quota checked at boot time.

Additional Run-time Overhead

Whenever disk quotas are initially enabled for a file system or are re-enabled after having been disabled with “quotaoff(1M)”, and whenever new, non-zero, quotas are established on a file system for a user or group of users, quotacheck must be re-run on that file system. This incurs the same overhead as mentioned under System Boot, above.

Disk Space Requirements

Disk Quota commands are contained in the DQUOTA files.

300 45 Kbytes

800 About 120 Kbytes

/quotas file

Each file system for which quotas are enabled contains a “quotas” file located at the root of that file system. This file is sparsely filled with per-user quota information. The reported size of the file, and thus its size if backed up and restored, is equal to 32 bytes times one plus the highest uid for which quotas are established. For example, if the highest user id with quotas on “/” is 2034, then the “/quotas” file will have size $32 \times (1 + 2034) = 65120$.

/etc/mnttab, /etc/checklist, getmntent(3)

Two new options values (“quota” , “noquota”) are supported.

_HP OSF/Motif

See information in the topic named “_X11”.

_HP-UX Clusters

A Cluster and Kernel Memory

The diskless subsystem dynamically allocates kernel memory for use by remote requests. The default amount reserved is appropriate for a cluster root server system, but not necessarily for clients. Follow the examples below for optimal memory usage. These rules should be applied to the uxgen input file (usually “/etc/conf/gen/S800”) on Series 800 systems and applied to the config input file (usually “/etc/conf/dfile”) on Series 300/400 systems.

Standalone System

- The diskless subsystem should NOT be included. Be sure the following line does not exist or is commented out.

```
include dskless;    /* S800 */
```

or

```
dskless    /* S300/S400 */
```

Diskless client with small amount of memory

- The diskless subsystem should be included. Add the line:

```
include dskless;    /* S800 */
```

or:

```
dskless    /* S300/S400 */
```

- The tunable parameter “server_node” should be set to zero. In the tunable parameter section add:

```
server_node 0;    /* S800 */
```

or:

```
server_node 0    /* S300/S400 */
```

Client with Locally Mounted File System

- The diskless subsystem should be included. Add the line

```
include dskless;    /* S800 */
```

or

```
dskless    /* S300/S400 */
```

- The tunable parameter “server_node” should be set to one. In the tunable parameter section add:

```
server_node 1;    /* S800 */
```

or:

```
server_node 1    /* S300/S400 */
```

- The tunable parameter “num_cnodes” should be set to one. In the tunable parameter section add

```
num_cnodes 1;    /* S800 */
```

or

```
num_cnodes 1    /* S300/S400 */
```

Server

- The diskless subsystem should be included. Add the line

```
include dskless; /* S800 */
```

or

```
dskless /* S300/S400 */
```

- The tunable parameter `server_node` should be set to one. In the tunable parameter section add

```
server_node 1; /* S800 */
```

or

```
server_node 1 /* S300/S400 */
```

- The tunable parameter `num_cnodes` should be set to the number of clients this system serves. In the tunable parameter section set `num_cnodes` accordingly.

Locally Mounted File Systems

Summary of Change

Locally Mounted Filesystems consists of two new features to the HP-UX Clustered Environment. First, it provides Auxiliary File Servers which allow the user to mount a file system that resides on a disk that is physically attached to a cluster node other than the root server node. Second, it provides Auxiliary Swap Servers. This allows non-root-server nodes to act as swap servers for client nodes in the same cluster.

Machines Affected

Only machines which are part of the HP-UX Clustered Environment are affected.

Features

- HFS mount on an auxiliary server

A device containing an HFS type of file system may be mounted onto a mount point on the cluster root server's HFS type file system, even if the device being mounted is physically attached to a non-root server cnode. A local disk may be used for both swapping and file system mounts if it is so configured.

■ **Restrictions to this feature:**

- All commands which act on a device such as `mount(1m)`, `fsck(1m)`, `makefs(1m)`, `mediainit(1m)`, `newfs(1m)`, and `umount(1m)` must be executed on the cnode where the device is physically attached.
- The mount point for an HFS or NFS mount must reside on a device which is physically attached to the root server.
- HP-UX system software must be installed only onto devices connected to the cluster's root-server cnode.

LMFS and NFS

Auxiliary servers with locally mounted file systems may also be NFS servers.

Global visibility

A file system mounted on an auxiliary server is globally visible to all nodes in the cluster and has the same appearance to the user as would a file system mounted on the root server node. This is true for NFS mounts as well. The `mount(1m)` command with no arguments will show all file systems mounted on all nodes in the cluster and will give the same results when run on any node.

/etc/mnttab

Device file pathnames in “`/etc/mnttab`” will be expanded to include their hidden (CDF) components so that devices with similar names on different cnodes may be distinguished in the global mount table.

Shutdown(1m) and reboot(1m)

The “`shutdown(1m)`” and “`reboot(1m)`” commands will unmount any locally mounted file systems prior to rebooting cnodes. When a “`reboot(1m)`” command is run on an auxiliary swap server, the swap server must force the shutdown of all of its swap clients.

3-44 Part 2: Release-specific Topics

Impact

- Kernel and system call interface changes

- mount(2)

```
mount(spec, dir, rwflag);
char *spec, *dir;
int rwflag;
```

In previous releases when mount(2) was called from a diskless node, “spec” was interpreted as a device attached to a cluster server. In 8.0 if the device referred to by either the “spec” argument matches a device that is physically connected to the cnode which is executing the system call, an HFS filesystem on that device will be mounted on as part of the cluster’s global file system hierarchy. It is recommended that the “spec” and “dir” arguments always be fully expanded pathnames.

- vfstmount(2)

```
vfsmount (type, dir, flag, data);
int type;
char *dir;
int flags;
caddr_t data;
```

Before 8.0 an attempt to mount an HFS file system with vfstmount(2) call from a diskless node would fail with an EINVAL error. In 8.0 if the device referred to by either “spec” or “data” argument matches a device that is physically connected to the node which is executing the system call, an HFS filesystem on that device will be mounted on as part of the cluster’s global file system hierarchy.

- umount(2)

```
umount(name)
char *name;
```

Before 8.0 when umount was called from a diskless node and “name” referred to a block-special file, “name” was interpreted from the root server. When path referred to a directory where a HFS file system was mounted, an EINVAL error was returned. As of 8.0 the device referred to (either directly by the device file pathname or implicitly by the mount point directory) by

“name” must correspond to a valid local device, and if a device name is specified, the device file’s cnode id must match the cnode id of the executing system call.

■ `ustat(2)`

```
ustat(dev,buf)
dev_t dev;
struct ustat *buf;
```

The `ustat(2)` system call returns file system information about a mounted file system. The user provides a device number to specify the file system for which information is desired. Because mounts are global but device numbers are not unique through the cluster, `ustat(2)` may not always know which file system is referred to by a device number. To remove this ambiguity, HP-UX provides an encoded device number, obtained from the “st_dev” field of the “stat” structure which is returned from `stat(2)`. This encoded number will refer uniquely to the device regardless of which cnode is making the system call. The recommended method for obtaining information about a mounted file system is through the `statfs(2)/fstatfs(2)` system calls.

■ `getmount_cnt(2)`

```
int getmount_cnt(mtab_time)
time_t *mtab_time;
```

This system call is new to 8.0 and it is used to keep “/etc/mnttab” up to date. It returns the number of currently mounted file systems and if “mtab_time” is not a null pointer, it sets the value pointed to by “mtab_time” to the time of the last status change in the kernel mount table. It will remain undocumented for a few releases so that the interface may be changed if it is found inadequate.

■ `getmount_entry(2)`

```
int getmount_entry(index, fsname, mntdir, md)
int index;
char *fsname;
char *mntdir;
struct mount_data *md;
```

Like “`getmount_cnt(2)`”, “`getmount_entry(2)`” is also used to keep “/etc/mnttab” up to date. It returns information about the mounted file

system that is referred to by “index”. Each consecutive integer value of “index” from zero to one less than the return value of a “getmount_cnt(2)” call corresponds to a mounted file system.

Mntdir	points to a path name up to PATH_MAX bytes long representing the directory the file system is mounted on.
Fsname	points to a string up to PATH_MAX bytes long that describes the device special file of the mounted file system.
Md	is a structure of type “mount_data” which is defined in <sys/vfs.h>.

Crash recovery

Once a cnode has been declared dead by the cluster via the standard HP-UX cluster state-of-health mechanism, the file systems belonging to that cnode will effectively be unmounted from the file hierarchy and all subsequent accesses by other cnodes to the formerly mounted file system may result in errors.

Open files

Operations on open file descriptors which only affect data structures local to the client will succeed (i.e. dup(2), dup2(2), lseek(2) in the SEEK_SET and SEEK_CUR cases and fcntl(2) with the F_DUPFD, F_GETFD, F_SETFD, F_GETFL, F_SETFL operations). Whereas operations which need to go over the network for one reason or another will fail (i.e. ftruncate(2), fchown(2), read(2), write(2), fstat(2), and lseek(2) in the SEEK_END case).

FIFO's

Open FIFO's will be closed and all operations that were blocked on FIFO's will unblock.

Current working directories

Processes whose current working directories were located in a crashed file system will effectively have undefined current working directories until a chdir(2) call to an absolute pathname re-roots the process in a valid directory.

Sockets

Unix domain sockets that are bound to a path name that refers to a crashed file system will continue to work. Subsequent bind(2) operations to the same path will fail.

File Locks

All file locks on an open file will be released if the server for that file crashes.

NFS mounts

NFS mounted file systems are not recovered even if the cnode that initiated the mount crashes, since NFS mounts are a cluster-wide resource and do not really belong to any one cnode. Once the mount is completed, every cnode has the identical client state for an NFS mount.

Performance data

Performance of locally mounted file systems on non-root server cnodes should be equivalent to the performance of file systems mounted on the root server cnode. Exceptions to this are the mount(1m), umount(1m), reboot(1m), and shutdown(1m) commands which will be slightly slower due to the need for more remote cnode interaction.

Size requirements

For an auxiliary server kernel, the current diskless kernel configuration should be modified to include the appropriate disk driver. Adding a disk driver to an S300 kernel may increase the size of the kernel by approximately 32k bytes. Adding a disk driver to an S800 kernel may increase the size of the kernel by 176k bytes.

Support for auxiliary servers in HP-UX clusters

Summary of Change

The addition of auxiliary servers in HP-UX clusters provides two new features to non-root-server cnodes:

- Auxiliary File Servers

3-48 Part 2: Release-specific Topics

This feature allows the user to mount a file system which resides on a disk that is physically attached to a cluster node other than the root server cnode. This feature impacts the following HP-UX system commands, libraries and files:

```
mount(1m), umount(1m), reboot(1m),  
shutdown(1m), getmntent(3x), df(1m),  
bdf(1m), sam(1m), update_mnttab(3x),  
/etc/checklist, and /etc/mnttab
```

■ Auxiliary Swap Servers

This feature allows non-root-server cnodes to act as swap servers for client cnodes in the same cluster. This feature impacts the following HP-UX system commands and files:

```
reboot(1m), shutdown(1m), sam(1m),  
rbootd(1m), /etc/clusterconf
```

Information

For more information on auxiliary servers see the Locally Mounted File System ES or the HP-UX 8.0 Commands ES.

Configuration

Cluster clients which have a physically local mass storage device may make that device available for use by the entire cluster for swap space and/or file system space.

Features

The “mount(1m)” command will have two additional options for displaying information about local mounts. The “-l” option will provide information about local mounts only; the “-L” option will provide information about local mounts and (cluster wide) NFS mounts only. The “-l” option will also be made available for “bdf(1m)” and “df(1m)”.

The “shutdown(1m)” and “reboot(1m)” commands will be modified to:

- Shutdown/reboot any swap clients when a swap server is shutdown/rebooted.
- Umount any locally mounted file systems prior to shutdown/reboot.

- The “mntent” structure returned by “getmntent(3x)” will include an additional field which will indicate the cnode on which a special device file for a specific file system mount entry resides.
- “/etc/checklist” and “/etc/mnttab” will display all special device file pathnames and all file system mount point pathnames as fully expanded CDFs to remove any ambiguities that might be introduced in “/etc/mnttab” due to hidden directories.
- “rbootd(1m)” will be modified to prevent a swap client from booting before its swap server has booted.
- Entries in “/etc/checklist” may specify a swap server other than the root server or the local cnode.

Impact

There is a potential for impact on applications which assume that all mounted file systems in a cluster must reside on the root server cnode.

_HP-UX Commands

New Commands for 8.0

There are several new commands provided to help identify the architecture which a script is running on. These new commands are located in “/bin”. The following table describes their behavior on different architectures along with similar commands provided in the past. New commands are marked with an asterisk (*).

COMMAND	S300/S400	S700	S800/S600
/bin/hp9000s300	TRUE	FALSE	FALSE
* /bin/hp9000s700	FALSE	TRUE	FALSE
/bin/hp9000s800	FALSE	TRUE	TRUE
* /bin/hp-mc680x0	TRUE	FALSE	FALSE
* /bin/hp-pa	FALSE	TRUE	TRUE

Note that “/bin/hp9000s800” will currently return true on a Series 700. This may change in the future and it is recommended that scripts not rely on this behavior. The “/bin/hp-pa” command should be used if the script expects a true value for the Series 800, 700 and 600. The “/bin/hp9000s700” command should be used if the script expects a true value only for the Series 700.

The AWK Command

Summary of Change

In 8.0, “awk” has been upgraded to the version described in the “The AWK Programming Language”. Because of this change, certain incompatibilities have been introduced between the new (8.0) and historical (7.0) awk:

Lvalues referencing awk fields

POSIX 1003.2 states that an <lvalue> (any entity that can be assigned to (i.e., on the left side of an assignment operator)) which begins with a “\$” references a field.

Example:

```
{ x=0 ; $0="old"; $x="new" ; print $0 }
```

If you use historical awk on this example, it will print out the word “old”. The new awk will interpret “\$x (with x=0)” as referring to “\$0” and will set “\$0” to the string “new”.

In either awk, a “\$” does not have to precede a variable name to allow you to extract its value. Awk is not the shell.

New reserved words

Using reserved words as <lvalue>’s will cause errors. The following functions are new in the 8.0 version of awk:

return	atan2	cos
sin	rand	srand
match	sub	gsub
close	getline	system
function		

The entire list of reserved words is:

if	while	do
for	delete	break
continue	next	exit
return	print	printf
in	atan2	cos
sin	exp	log
sqrt	int	rand
srand	index	length
match	split	sprintf
sub	substr	close
gsub	getline	system
function		

The following pre-defined variables are new to the 8.0 awk:

ARGC	ARGV	FILENAME
FNR	OFMT	RLLENGTH
RSTART	SUBSEP	

The entire list of pre-defined variables is:

ARGC	ARGV	FILENAME
FNR	FS	NF
NR	OFMT	ORS
ORS	RLENGTH	RS
RSTART	SUBSEP	

Specifying input field separator (FS)

The historical version of awk allowed you to use the -f option to specify the input field separator (FS) as well as specifying the file awk should get its script from. The new awk no longer supports this. You must use the -F option when specifying FS, and the -f option when specifying an awk script.

Extended regular expressions

The new awk supports POSIX extended regular expressions.

Non-existent fields

Assignment to non-existent fields will create both the field in question and all intermediate fields from $\$(NF+1)$ on.

Increasing the value of NF directly will also create all intermediate fields between the old and current values of NF.

Example:

```
echo "one two" | awk '{$3="three"; print}'
```

The new awk will produce:

```
one two three
```

The old awk will produce:

```
one two
```

Configuration

Changes apply to both series 300 and series 800.

Impact

Existing awk scripts will need to be re-written to support the syntax and semantics of the new “awk”.

Alternatives

For backward compatibility reasons, the historical version of awk is available as “/usr/old/awk-7.0”. This version is obsolete and may not be shipped in the future.

The CAT Command

Summary of Change

A new option was added, “-c”, to coalesce three or more adjacent <newline> characters to two <newline> characters.

The CHGRP CHMOD CHOWN Commands

Summary of Change

A new option was added, “-R”, to recursively descend the file tree and complete the change requested.

Alternatives

Find(1) should still be used for special consideration of directories vs files.

Performance data

This functionality is mandated by POSIX.2.

The COMM Command

Summary of Change

This command will now parse “-23” and “-2 -3” correctly and the same.

The DD Command

The “dd(1)” command still discusses “blocks”, not “records”. POSIX.2, Draft 9 requires it to count “records”. This is still not a resolved issue with POSIX.2, Draft 9. An objection has been filed with POSIX.2.

Alternatives

Don’t parse the command output looking for the words “records” or “blocks”. Instead, use positional offsets.

The ED Command

Summary of Change

A new option was added, “-s”, which is the same as the “-” option, which is deprecated. The optional “-s” suppresses the printing of character counts by “e”, “r”, and “w” commands, of diagnostics from “e” and “q” commands, and of the “!” prompt after a “!shell” command. See how “env” uses “getopt”.

Impact

The “-” option will be deleted in a future release.

The ELM Command

In 7.x releases, “elm” expanded TAB to SPACE when you sent a message; that is, the default was ON.

In the 8.0 release, the default is changed to OFF, but you can turn it to ON in the options editor.

In the 8.0 release, there is no “.elm” folder, but “elm” asks if you want to create one.

The FIND Command

Summary of Change

New options are implemented, “-nouser”, “-nogroup”, “-nodevcid”, “-linkedto” and “-xdev”.

Features

-nouser

True if file belongs to a user ID with the equivalent of “getpwuid(1)” returning NULL (not a valid user).

-nogroup

True if file belongs to a group ID with the equivalent of getgrnam(1) returning NULL (not a valid group).

-nodevcid

True if file is a block or character special file that has a cnode id that is not in /etc/clusterconf.

-linkedto

True if file is linked to the given file.

-xdev

Always true; causes find not to continue descending past directories that have a different device ID.

The FSCK Command

Summary of Change

Fsck is invoked on the root file system before running init.

Configuration

A new startup file will be run before “/etc/init”, and will be responsible for checking the file system before allowing init to run.

Features

This feature will prevent multiple reboots after a crash by allowing fixes to the root file system to be incorporated in the running kernel.

Impact

Single-user systems will now run checks on their file systems during system startup.

Alternatives

The kernel variable “skip_fsck” can be set to disable this feature. If this variable is set, the system will behave as before.

The GETCONF Command

Summary of Change

New command. Get system configuration values.

Features

Dynamic and static system configurations are available by calling “pathconf(3C)”, “strconf(3C)” and “sysconf(3C)”.

Impact

Header files limits.h and unistd.h were changed to support POSIX defined names and values.

The GREP EGREP FGREP Commands

Summary of Change

Behavior may be different when grepping files with long lines. Formerly a match would be attempted only on the first BUFSIZ characters of a line; if a match was found, the whole line would be printed.

Now a match is attempted on the first LINE_MAX characters. LINE_MAX is a POSIX.2 required configuration value which normally defaults to 2048. This

change means that some lines that did not match in 7.0 and before will now match.

Features

The configuration value of `LINE_MAX` can be had via `“sysconf()”`.

Impact

Should be minimal and positive, some matches will now be found that were not in earlier releases.

The ID Command

Summary of Change

The `“id(1)”` command now outputs supplementary group ids, if any.

Impact

This change may impact usage of `id(1)` in scripts if the output is parsed. If a script somehow depends on a fixed number of quantities being returned by `id(1)`, behavior could change.

Alternatives

Depending on the options, do one of the following; parse for `“gid=”` or `“uid=”` if full output is expected. If expected output is limited to user or group id, only examine first field of returned data.

The KSH Command

Summary of Change

In 8.0, `“ksh”` has been upgraded to the version described in the book `“The Kornshell Command and Programming Language.”` This version of `“ksh”` is known as `“ksh88”` (11/16/88b).

Features

The major new features provided by “ksh88” are listed below.

Additional pattern matching operators.

```
?(pattern-list)
*(pattern-list)
+(pattern-list)
@(pattern-list)
!(pattern-list)
```

“expr” is built-in, designed to replace test.

Additional test operators:

```
-0 file      true if file owned by effective user
-G file      true if file owned by effective group
-S file      true if file is a socket
-a file      true if file exists
-o option    true if option is set
```

Expanded integer arithmetic. New operators are:

```
&&, ||, &, |, ^, <<, >>, ~.
```

Also, all operators of the form “+=” are recognized by “ksh88”.

“fbgetopts” built-in.

New parameters are: BERRNO FPATH LINENO OPTARG OPTIND PS4

Aliases can expand in other aliases.

New options to the built-in and new traps: DEBUG and CHLD

“umask” accepts symbolic masks (like chmod)

New I/O redirection operators:

```
<>, >|, >&p, <&p.
```

Support for multiple co-processes

The “pwd” and “cd” built-ins accept options to change the way symbolic links are handled.

Compatibility Issues

The meaning of the “-p” option to the set built-in has been changed as well as its name. In the previous version, “-p” caused PATH to be reset to a default value and the shell not to run your personal “\profile” and “\$ENV” file. Instead, the shell executed a file named “/etc/suid_exec”. The “-p” option was enabled automatically whenever the real and effective user or group id were not equal. This was called the protective option. With the new release, the “-p” no longer resets the path to a default value. In addition, the effective user id and group id are set to the real user id and group id unless the shell has been invoked with the “-p” options which is now called privileged. In addition, turning off privileged, (set “+p”) sets the effective ids to their real values.

The typeset built-in no longer supports the “-e” and “-p” options.

Since RFA is no longer supported in 8.0, the netunam built-in has been removed.

The LN Command

Summary of Change

A new option was added, “-i”.

Features

Option, “-i” will inquire before overwriting an existing file.

Impact

This functionality is mandated by POSIX.2.

The LP Command (Spooler)

Summary of Change

Adding getopt broke backwards compatibility from common, tho undocumented usage.

Impact

Backwards compatibility is changed. Where previously undocumented usage allowed options to be specified after the file name, that is no longer the case. According to POSIX.2, Utility Syntax Guidelines 2.11.2, all options must occur before any operands. eg: “lp foo -n3” will not work, “lp -n3 foo” will work as expected.

The LS Command



Summary of Change

When the “-t”, “sort by modification time” option is given, filenames are sorted alphabetically after being sorted by time.

Impact

In cases in which multiple files in a directory have the same modification time, the order in which those filenames are displayed will probably be different from what it is in 7.0 and before. This will have no impact unless applications use the “-t” option and depend on the order in which filenames are displayed.

The MAKE Command

Summary of First Change

The “make(1)” command now ignores the SHELL environment variable. This is a POSIX.2 requirement.

Features

A “SHELL=<some/shell>” macro may be used within a makefile to select a shell in which to execute “make(1)” command lines. Otherwise, “make(1)” will default to “/bin/sh”.

Impact

This will impact usage in which the value of the user’s SHELL environment variable is implicitly relied upon for correct behavior of the command lines

in a makefile. In such cases, makefiles must be altered to include a macro assignment of the form:

```
SHELL = /my/shell.
```

Alternatives

Makefiles could also be altered to work as expected with the default, “/bin/sh”.

Summary of Second Change

The “make(1)” command will now execute the rules associated with a given target if any of the target’s dependents have a modification time which is newer than or equal to the target’s modification time.

Impact

This change was made to fix a defect which showed up with faster processors. Archive libraries made from subdirectories of “.c” files were sometimes not complete because the compilations happened so fast that the last modification time of the archive library and the “.o” files were the same, and the archive step would not take place.

This change, however, has the potential of causing unnecessary execution of commands. Usually, this causes nothing more serious than a bit more execution time, but depending on the commands associated with the target, an error message may be generated, which will halt make unless the “.IGNORE” option is turned on for that command or for the entire file. For example, issuing “ln -f” will return an error if done on two files with the same modification time.

Alternatives

If a previously safe command now generates an error, the command can be preceded by a “-” which causes make to continue executing the makefile even if errors are generated by that line. If any errors occur, an error message from the shell executing the command will still appear on “stderr”, but make will not halt.

The MKBOOT Command

There are two new commands which allow users to install, update or remove boot programs from a disk device. They are the “mkboot” and “rmboot” commands.

The “mkboot(1M)” command allows the user to specify which boot program to install based on the desired architecture if necessary and the desired device to install this on. The following is the list of the boot programs installed by architecture.

```
/usr/lib/uxbootlf" is installed for the s800 and s600
/usr/lib/uxbootlf.700" is installed for the s700
/etc/boot" is installed for the s300 and s400
```

The “mkboot” command also provides an option to allow the user to increase the space required by the boot programs. This option may be needed in the future to provide for an increase in the boot program size.

The “rmboot” command removes the boot programs from the boot area if the given device identifies a Series 300, 400 or 700. No action is taken by “rmboot” for the Series 800 and 600 since this area is reserved anyway.

Please refer to the “mkboot(1M)” man page for more detail on “mkboot” and “rmboot”.

The MONITOR Command

A New Command

The “/usr/contrib/bin/monitor” command was unsupported in the 7.0 release. In this release, “monitor” is supported for Series 600/800, but is NOT supported for Series 300/400.

The RM Command

Summary of Change

A new option, “-R” was added, which is equivalent to “-r” (recursively descend the file tree).

The RMDIR Command

Summary of Change

Two new options were added, “-R” which is equivalent to “-r” (recursively descend the file tree) and “-p” to remove parent directories, if they become empty during a rmdir of a lower level directory.

The SORT Command

Summary of Change

Added new option, “-k keydef”, which is a sort key field definition.

Impact

Adding getopt broke backwards compatibility from common, though undocumented usage. Where previously undocumented usage allowed options to be specified after the file name, that is no longer the case. According to POSIX.2, Utility Syntax Guidelines 2.11.2, all options must occur before any operands. eg: “sort foo -obar” will not work, “sort -obar foo” will work as expected.

The TOP Command

A New Command

The “/usr/contrib/bin/top” command was unsupported in the 7.0 release. In this release, “top” is supported for Series 600/800, but is NOT supported for Series 300/400.

The UNIQ Command

Summary of Change

Added new options, “-f” and “-s” for skipping fields and characters

Impact

In a future release, the “-n” and “+n” options will be deleted, since they are being replaced by “-s” and “-f” respectively.

The UUENCODE Command**Summary of Change**

This utility was added and documented for the first time in HP-UX.

Impact

This utility was added because the C product is no longer part of the core HP-UX system. The “shar(1)” utility requires the ability to encode a message.

The UUDECODE Command**Summary of Change**

This utility was added and documented for the first time in HP-UX.

Impact

This utility was added because the C product is no longer part of the core HP-UX system. The “shar(1)” utility requires the ability to dncode a message.

_HP-UX Kernel

Autoconfiguration/800

Summary of Change

The Series 800 kernel now automatically configures devices during and after system boot, for the most part eliminating the need for hardcoded I/O configuration information in the “uxgen(1m)” input file.

Autoconfiguration is part of the core system in 8.0, so there is no single program or subsystem defining it. Numerous drivers have been enhanced to support the dynamic identification and configuration of their associated hardware. In addition, several commands, described under “Features” have been added or enhanced to manipulate the addition and removal of I/O hardware to and from a machine.

Configuration

This applies only to the Series 600/800.

Features

New and/or enhanced commands:

- | | |
|--------|--|
| uxgen | enhanced to support the defaulting of system special devices: console, root, swap, and dump. |
| insf | enhanced to support the addition of new devices to the system. |
| rmsf | a new command to remove special files or devices. |
| ioscan | a new command to display I/O system configuration. |
| ioinit | a new command to download I/O information into the kernel. |

All core drivers except those noted below have been enhanced to support autoconfiguration.

Non-Features

The core drivers “instr0”, “gpio0”, “gpio1” do not support automatic configuration in 8.0. Non-core drivers that do not support autoconfiguration

3-66 Part 2: Release-specific Topics

include “psi0”, “pdn0”, “rti0”, and “rti1”. These drivers must still be specified completely in the “uxgen(1m)” io statement.

Impact within the kernel

Drivers written outside of HP that do not change should continue to function, although they will not be configured automatically. To take advantage of autoconfiguration, a driver must make its data structures dynamic, add support for new probe request messages, and define an identify procedure. This process is documented in the “Series 800 Autoconfiguration External Specifications” .

Although care was taken to ensure that old drivers would still work, there are some visible changes to the I/O system. They include:

- For backward compatibility, the existing “iotree” has been retained. However, it only contains those drivers statically configured in the “uxgen(1m)” input file, not those devices automatically configured.
- The “iotree” is obsolete, and will only be supported through Release 8.0.
- The “lu” field in the “iotree” is not usable outside the kernel, and may have no relation to the device’s actual logical unit.
- “Io_get_mem()” will no longer panic if there is insufficient memory to meet a memory request. Rather, it will return a null pointer.
- The function “port_to_index()” is no longer the method by which to obtain port information, and it will be removed sometime after Release 8.0; it has been superseded by a function called “io_port_info()”.
- The parameter to the driver “attach()” routine, which is an index into the static “iotree”, will be valid only if this manager instance is present in the “uxgen(1m)” io statement.

Impact outside the kernel

The disk naming convention has changed to be consistent across all drivers. Instead of being named:

```
/dev/dsk/c 1u d0s*, /dev/dsk/c100 1u d0s*,  
and /dev/dsk/c200 1u d0s*
```

all disks are now named:

```
/dev/dsk/c lu d0s*
```

This is handled transparently through update, with symbolic links from the old naming convention to the new.

Several ad hoc mechanisms of determining the system I/O configuration are now obsolete.

- Opening “/dev/kmem” and seeking through the I/O internal data structures (an unsupported operation) is guaranteed to return invalid data.
- The file “/etc/devices” no longer exists.
- The format of the “uxgen(1m)” input file has changed, particularly in the system device specification and the I/O configuration statement. Programs which scan “/etc/conf/gen/S800” will likely fail.

Alternatives

A new supported command on the Series 800, “ioscan(1m)”, prints information about the system’s I/O configuration, and supersedes “/etc/devices” and “/dev/kmem” for extracting hardware information. It is documented in its associated manual page. In addition, there is an HP-internal-only library to extract additional I/O information from the running kernel.

Greater Than 60 Open Files

Summary of Change

Each pre-8.0 process had a limit of 60 open files. Now, a mechanism is provided for a process to dynamically configure the number of open files it has.

There are two new tunable system limits on the number of file descriptors per process, a soft limit which a process can change by using the “setrlimit()” system call and a hard limit which can only be exceeded by super-user processes. The soft limit represents the maximum number of open files a process may have without issuing the “setrlimit()” system call. A processes’ soft limit may be increased up to the hard limit or decreased to any value which is greater than the number of the highest file descriptor allocated. Only

a super-user process may increase its soft limit beyond the hard limit. For performance reasons neither the hard or soft limit may exceed 2048.

Configuration

The tunable parameter “maxfiles” represents the soft limit and “maxfiles_lim” represents the hard limit. Both parameters default to 60, so in order to allow non-root-processes to utilize this feature, “maxfiles_lim” must be set at kernel generation time to a value greater than “maxfiles”. It may be necessary to increase the value of “nfile”, which represents the number of open files system wide, when configuring in this feature.

Features

Using this feature gives a process the capability to increase or decrease its limit on open files. This is useful for applications which require more than 60 open files.

Impact

Since there is no specified limit on the number of open files a process may have, NOFILE, which is defined in “/usr/include/sys/param.h”, should no longer be used as the maximum number of open files.

Alternatives

Software which currently uses NOFILE should instead use the “getnumfds()” system call. For example, loops which are written:

```
for (i = 0; i < NOFILE; i++) {  
    /* do something to file */  
}
```

should be replaced by:

```
num = getnumfds();  
for (i = 0; i < num; i++) {  
    /* do something to file */  
}
```

NOFILE will always be set to the system default soft limit of open files, which for 8.0 is 60. Unless a process opens or inherits more than 60 files, the code which continues to use NOFILE will not break.

Another example is the “select()” system call. The potential problem is with calls to “select()” where instead of specifying the exact number of file descriptors to be examined, the caller uses “NOFILE”. As long as the number of open files the caller has is less than or equal to “NOFILE” this won’t break. A problem will occur when a process opens more than “NOFILE” files and wishes to select on them. In this case the masks for “select()” should be defined large enough to allow for one bit per file descriptor. Internally the masks for select are defined large enough to accommodate the possibility of 2048 file descriptors since there is a system wide absolute limit of 2048 open files per process.

For the case where the number of open files a process wishes to select on is greater than NOFILE and not known, a macro is defined in user.h to compute the number of words necessary for the bit mask argument to select.

```
#define NWORDS(n) (((n) % (SFDCHUNK)) == 0 ? (n >> 5) : ((n >> 5) + 1))
```

Now, a call to “select()” such as the following code:

```
int fds;
int mask1[2];
int mask2[2];
int mask3[2];
struct timeval *timeout;
...

select(fds, mask1, mask2, mask3, timeout);
...
```



could be written as:

```
int fds;
int *mask1;
int *mask2;
int *mask3;
struct timeval *timeout;
mask1 = (int *)malloc(sizeof(int) * NWORDS(getnumfds()));
mask2 = (int *)malloc(sizeof(int) * NWORDS(getnumfds()));
mask3 = (int *)malloc(sizeof(int) * NWORDS(getnumfds()));
fds = getnumfds();
...

select(fds, mask1, mask2, mask3, timeout);
...
```

This will supply enough bits for the input masks to the “select()” call.

Header Files

< curses.h >

The < curses.h > header file has been modified so that it can be used with either the < termio.h > header file or the < termios.h > header file. In order to achieve this, < curses.h > no longer includes < termio.h >. Source code which included < curses.h > and used some of the structures, typedef's, macros or declarations included in < termio.h > will now have to explicitly include this header file.

Libraries Used by the Kernel (e.g. “srرتol()”, “closedir()”)

Shared Library Utilization (Core Commands and Libraries)

With the new Virtual Memory (VM) system being added to HP-UX 8.0, the ability to add a sound shared library implementation now exists. With the appropriate tools provided from LMO, compilers and linkers, the HP-UX core commands group can create shared libraries. This would allow HP-UX commands to be built without actually having shared library code copied into its text space. This would reduce the size of each of the executables.

Summary of Change

The shared libraries project will be providing several shared libraries for the HP-UX 8.0 release. “Libc.[a,sl]” the standard “C” library, “libm.[a,sl]” the standard math library, and “libcurses.[a,sl]” the terminal curses library will be provided in both shared (“libxx.sl”) and archived (“libxx.a”) format. The HP-UX commands will also be linked with these shared libraries with the exception of those necessary for maintaining and installing shared libraries.

The profiling libraries will not be provided in a shared format. This may be revisited for future releases if there is a demand for it. Since programs can still be linked with the archived versions of “libc.a” and “libm.a”, which both have profiled versions, complete profiling information can be gathered.

Configurations Supported

The above mentioned shared libraries and commands will be supplied on all supported HP-UX Release 8.0 platforms (i.e. S3xx, S8xx, and S6xx).

Lockable Memory

Availability on an 8 MB System

On a system with 8 mega-bytes of memory, if the default values of configurable parameters as specified in “/etc/master” and “/etc/conf/gen/S800” are used to generate the kernel, the system may not have enough memory available to be used as lockable memory. If a user needs to use lockable memory, it is recommended that the “maxusers” parameter be reduced to a value no greater than 16. The size of many kernel data structures are determined as a function

3-72 Part 2: Release-specific Topics

of “maxusers”. Reducing the value of “maxusers” will allow memory to be freed up and be used as lockable memory.

To reduce the value of “maxusers”, add the following line in the “/etc/conf/gen/S800” file and remake a kernel:

```
maxusers XX;
```

Substitute XX with the new value chosen for “maxusers”.

Possible Problem with Memory Locking

When a process attempts to lock down some of its memory, whether by calling “plock()” or “shmctl()”, the system first checks to see if the system can possibly ever fill the request. If not, the call returns an error right away. If there could (sometime) be enough memory the system gathers it together, locking it down a page at a time. If at any time before it gets everything it wants, free memory goes to zero, the process blocks until more memory is freed. This continues until the system has locked down all the memory the process requested, at which time the system call completes and control is passed back to the user’s process.

The problem is that the system’s idea of the maximum lockable memory that might become available can be a little high, so that a process that tries to lock down the rest of the lockable memory could pass the “enough ever” test and go into the page at a time collection loop when there will never be enough to finish the request. If this happens, the process will block, with some or all of that memory locked down, and not make any forward progress. Depending on the circumstances, there may or may not be enough memory left for other processes to make forward progress.

This situation is not very likely to happen with real work loads because most processes don’t lock down any memory. Those that do would have to lock just the right amount to expose this problem. Too much and the call is refused with no hang, too little and the request is granted with no problem.

Shared Libraries

Summary of Change

8.0 introduces shared libraries into HP-UX. Changes were made in the virtual memory management system for process space layout and minor changes were made to process management.

Configuration

Whether a process uses shared or archive libraries is determined by a linker option. By default shared libraries will be used where they exist.

Features

See the Programming on HP-UX HP9000 Computers Manual for more information on Shared Libraries.

Impact

Shared libraries that have the file system mode of "r-xr-xr-x" with no ACL are assigned a public protection id on the Series 800. That means that once they are mapped into the memory management system for one process, it is available to everyone.

Size Requirement

With archived libraries, a process mapped in all the data required by the modules of that library that were required by the process. With shared libraries, each process that uses a shared library maps in all of the data contained in that library. In some conditions this may lead to additional swap space requirements for your system.

System V IPC

Summary of Change

In order to conform to the XPG3 standard, the definitions of a few IPC data structures as defined in <sys/ipc.h>, <sys/shm.h>, <sys/sem.h> and

<sys/msg.h> have been changed in the 8.0 release of HP-UX. The data structures and system calls affected are:

```
struct ipc_perm - affects shmctl(), semctl() and msgctl()
struct shmid_ds - affects shmctl()
struct semid_ds - affects semctl()
struct msgid_ds - affects msgctl()
```

Features

Three new system call numbers have been added for “shmctl()”, “semctl()” and “msgctl()” in the 8.0 release of the C library (“libc.a” and “libc.sl”) in order to work with the new definitions of the structures.

Impact

Object code compatibility is supported for “a.out” files (executables) made before release 8.0. Therefore, programs made before 8.0 will continue to work without requiring recompilation. However, object code compatibility is not supported at the “.o” files (relocatable object files) level. In other words, “.o” files that reference the data structures and system calls mentioned above should be recompiled with the 8.0 header files if they are to be linked with the 8.0 “libc.a” or “libc.sl”.

Swapping Subsystem

Summary of Change

At 7.0, support for dynamically adding file system swap was added to the swapping subsystem. Dynamic swap allocation was completed in 8.0 by adding the functionality of allocating swap devices to a running system. The ability to assign priorities to these devices was also added. In addition, improvements were made in the allocation of swap space to resolve problems with fragmentation which led to unusable swap space.

Configuration

Two new tunable parameters “nswapdev” and “swchunk” have been added to the swapping subsystem. The parameter “nswapdev” is the number of swap devices that may be configured into this system, either statically at kernel configuration time, or dynamically at run-time. The system default

for this parameter is “10”. The parameter “swchunk” is the size of each contiguous block of swap space to be allocated to the system for swapping. The default is “2 megabytes”. Additionally, the meaning of the tunable parameter “maxswapchunks” was extended to the maximum number of chunks of swap space the kernel will be able to use both in device swap and pm filesystem swap. The expression “swchunk * maxswapchunks” is the total amount of swap space that the kernel is able to use. More space may be configured into the system, but it would be wasted until the system was reconfigured and rebooted with a new value for “maxswapchunks”.

For more information on configuring your swapping subsystem please see the “System Administration” documentation.

Virtual Memory

Summary of Change

Operating-system code for HP-UX virtual memory has been improved; changes affected page layouts and how processes and pages of data are scheduled and written to and from memory.

Configuration

The configuration of the virtual memory management system has not changed except for obsoleted tunable parameters outlined in the Impacts section below and the addition of tunable parameters for the swapping subsystem that are detailed in the Swapping portion of of this document.

Features

■ “Dynamic_Allocation_of_Kernel_Memory”

In 8.0, many kernel data structures used for processes and swapping were reworked to dynamically allocate necessary space as needed. (That is, they reserve memory only as needed.) Because of this implementation, available memory is shared by user processes and the kernel, tying up less memory solely for kernel use. This means that in a general sense, more memory is available for user processes.

■ “Copy_on_Write”

3-76 Part 2: Release-specific Topics

The system used to copy the entire data segment of a process every time the process forked. “fork” time increased as the size of the data and stack segments increased. Because processes can be larger than available memory, copy time increased exponentially, causing system performance to degrade.

HP-UX implements copy-on-write, which enables the system to create processes more quickly. When a process forks, the parent process (consisting of several virtual images), creates a duplicate image of itself as child virtual addresses. But instead of actually copying the image, the parent pages are marked “copy-on-write.” The pages continue to reside in the parent address space, and are not copied until the process actually writes on the page. At that point, the old page is copied to a new page and the process gains access to the new page.

- “Larger_Amounts_of_Physical_Memory”

The amount of physical memory supported on the Series 800 increased from 256 megabytes to 768 megabytes at the 8.0 release.

- “Size_of_the_Buffer_Cache”

The size of the buffer cache used to be limited by the size of a system data structure. That limitation has been removed from the system, and the buffer cache is now limited only by the size of physical memory.

- “Sparse_Address_Space”

Changes made to the Virtual Memory Management System allowed the Series 300 to remove the restriction that each process’ page tables had to be contiguous. This made it possible to allocate page tables for a process’ address space sparsely. This results in a more efficient use of memory.

Impact

- “Fork_and_Vfork”

With the implementation of copy-on-write, the new VM system no longer supports “vfork”, the fast fork. Applications that call “vfork” will now use the same path as those calling fork and the semantics of the call will be the same. Applications that relied on the non-supported feature of passing information from the child to the parent by writing in the parent’s data area before “exec” or “exit” will break.

■ “Zombie_Processes”

A zombie process is a process that has terminated whose parent process has not yet waited for it to terminate via `wait(2)`. In the old VM system, zombie processes occupied a slot in the “proc” table, but any space that was allocated to it was freed. The new VM system, does not free certain parts of zombie processes until the parent waits for the child to terminate. This results in greater amounts of memory being taken up by zombie processes.

■ “Changes_in_Configurable_Parameters”

The tunable parameter “n`text`” is obsoleted in the new virtual memory management system. This was used in the old implementation as the size for the text table. The new implementation does not require this parameter, as the data structures that hold information about a process’ text are dynamically allocated.

The Series 300 tunable parameter “sh`mmaxaddr`” has been obsoleted. This parameter allowed the system manager to choose an address at which to start attaching shared memory when requests did not specify an address. Because it was a tunable system parameter, this address was the same for all processes on the system. It was necessary to make this tunable because there was a memory usage trade-off involved in assigning the attach addresses of shared memory segments. As described above, the old virtual memory management system expected the pte’s to be contiguous. Thus all pte’s from the end of the data segment to the beginning of the shared memory area had to be allocated even though they were unused. This wasted system memory if the difference of the addresses was high. If the shared memory attach address was chosen too low, a process may not be able to grow its data space because it collides with the area allocated for shared memory. The implementation of the sparse address space allows the system to choose an address for shared memory that is guaranteed not to collide with the data space of any process, since it is higher than the tunable parameter “ma`xdsiz`”.

_Installing HP-UX

S800 Install

Summary of Change

As of Release 8.0, CD-ROM and DDS are supported Install medium. All supported configurations of Series 800 systems can be installed from a CD-ROM or a DDS.

Impact

All currently supported install media continue to be supported as before. The interface to the CD-ROM and DDS is the same as the CTD.

808/815 HP-UX Release 8.0 Install/Update Notes

Due to increased functionality of 8.0 and its greater use of disk space, it has become necessary to document situations that involve Series 808/815 customers who will be performing Installs or Updates to HP-UX Release 8.0.

The situations the customer and/or HP support personnel should be aware of follow:

- Updating from HP-UX Release 7.0 to 8.0 with the Update program's "match what I have" feature or MU Runtime, using a 150Mb system disk.
- Installing HP-UX Release 8.0 MU Runtime, onto a 150Mb system disk.

Given the current file and fileset mappings within the Update program, more diagnostic filesets may be selected than you actually might need or even have space for. So you will need to make some decisions depending upon which of the above situations you fall under.

- After selecting the option to update existing filesets, examine your disk space situation using the Calculate Disk Space option and check to see if any ">>" are present.
- If not enough will be left over after the update finishes, then you will need to examine the Diagnostics partition to decide if more filesets are being selected than you actually need.

CORE-DIAG and HPIB-DIAG for system and main peripheral support.
NETWRK-DIAG for networking support.

- You may wish to also de-select any other filesets which may be selected, like INSTRUMNT-DIAG, GRAPHICS-DIAG or SCSI-DIAG, that you may have no need for and would improve your disk space situation considerably. If you can't do this, then you will need to contact your HP Support Person for more detailed help.

_Instant Ignition

Obsolete X Windows Environment for S 300/400

The X Windows Environment (fileset XENVIRON or XENV_CORE) based on HP-UX 7.0 and available for the Series 300/400 is no longer supported on HP-UX 8.0. Users of this environment are encouraged to switch to the Visual User Environment, a part of the X11-RUN fileset, and the IGNITION fileset. When the IGNITION fileset is loaded, the obsolete files in XENVIRON and XENV_CORE are removed. Refer to the manual “Installing and Updating HP-UX” for the Series 300/400 or the file `/etc/newconfig/Ignition/README`.

Ignition Fileset in S 300/400 Product

The IGNITION fileset has been added to the Runtime Product for the Series 300/400 (68K-based) Runtime Product. This fileset provides tools that configure the system to launch the Visual User Environment (VUE) when the system starts up. Usability tools such as `datebook`, `xcalendar`, `xdialog`, and `xhpcalc` are included in the IGNITION fileset. Refer to the manual “Installing and Updating HP-UX” for the Series 300/400 or the file `/etc/newconfig/Ignition/README` for more information on the IGNITION fileset.

_LP Spooler System

centronics (HP-PB)

Summary of Change

This is a new driver. The new driver is named “lpr2”.

Configuration

The centronics printer driver “lpr2” is added to the system by adding an include statement (for example, “include lpr2;”) in the gen file.

Features

This driver supports all the standard printer features. Both cooked and raw modes are supported plus io control.

The “slp” command may be used to perform io control on the printer.

Impact

There is no difference in functionality. In some applications a through-put increase may be seen.

Spooling on cnodes

Printers on cnodes can be configured via the new “-a” option for “lpadmin”.

_Online Diagnostics

Series 300 Computers

The Series 300 has Online Diagnostics

The Series 300 Core diagnostics contain the sherlock diagnostic system itself, and sherlock diagnostics which are considered to be globally useful.

OPDIAG

For testing the CD ROM drive.

GP3DDIAG

The 98720A Graphics Subsystem Diagnostic (GP3DDIAG) provides an on_line test of 98720A Graphics Subsystem on 300 SPU which supports the SHERLOCK diagnostic system. The Subsystem can be configured as the system console or a graphics peripheral. The type of tests that will be provided can test all functions of the 98720A Graphics Subsystem.

System Configuration and Dependencies

The diagnostic will work with a 98726A or 98725A graphics interface card.

GS3DDIAG

The 98730A Graphics Subsystem Diagnostic (GS3DDIAG) provides an on_line test of 98730A Graphics Subsystem on 300 SPU which supports the SHERLOCK diagnostic system. The Subsystem can be configured as the system console or a graphics peripheral. The type of tests that will be provided can test all functions of the 98730A Graphics Subsystem.

System Configuration and Dependencies

The diagnostic will work with a 98726A graphics interface card.

G98705DG

The G98705A|B|C Graphics Subsystem Diagnostic(G98705DG) provides an on-line test of 98705A|B|C Graphics Subsystem on the HP9000/300 or HP9000/400 processors running hp-ux release 8.0 or later and supporting

SHERLOCK diagnostic system. The 98705A|B|C Graphics Subsystem may be configured as the system console or as a graphics peripheral. The tests which are provided exercise all functions of the 98705A|B|C Graphics Subsystem.

System Configuration and Dependencies

The diagnostic will work on HP 9000/300 or HP9000/400 processors with remora interface card (98702A).

CE-UTIL

The Series 300 CE Utilities is a software package that provides facilities for testing, exercising, and diagnosing various peripheral devices and add-on products for the Series 300.

Utilities are provided for Optical storage (Optical), VME (VME), CS80 device exerciser (CS80), CRT adjustment (Crtadjust), EISA (EISA), and System Functionality Tests (Sft).

Enhancements and New Developments

SFT (System Functionality Tests): Support for 98627A RGB monitor, and digitizers (which only worked with the 98627A) deleted. The AGP/DGL libraries used by these tests to access the hardware are no longer supported by HP-UX 8.0. New Mux diagnostic added for the 98638A 8 Channel MUX. The Optical diagnostics modiag and acdiag have been improved and combined into one utility, oputil. An ESIA expander test has been added.

Files Added/Deleted

■ Deleted:

Sft/graphicstest, Sft/tablettest,
Optical/modiag, Optical/acdiag.

■ Added:

decustomize, Dmux/dmuxdiag, Dmux/help,
Dmux/helptext, Optical/oputil,
Optical/help, Optical/helptext,
EISA/eisa_test, EISA/help, EISA/helptext

3-84 Part 2: Release-specific Topics

SYS VERIFY

The HP-UX system verifier is an on line tool for end users that verifies system configuration, and re-configuration. It allows selective verification of the basic operation of HPIB and SCSI peripherals. Log files are generated.

Special Installation/Execution Instructions

- The executable files must NOT be stripped.
- Verify accesses the screen via curses, so the TERM environment variable must be properly set to match the terminal being used. After this is done, the tabs command should be run from the shell prompt. In general, if a terminal is configured properly for vi, it is properly configured for verify.

Enhancements and New Developments

Some additional signals are being caught/ignored. The user interface no longer waits for the user to enter CR in response to the “illegal reply” prompt. Instead, it displays the prompt for a fixed period of time, and then accepts normal input once again. Cleaned up the -u option to accurately print the syntax summary.

Known Problems

Please note that when verifying disks or cartridge tapes, the media must be inserted, and, in the case of disks, spun up. If these conditions are not met, the system could hang or crash, or the verifier could hang. There seems to be no way to test for these conditions programmatically. These problems may be driver related.

A SCSI CD ROM drive will not be recognized unless a medium has been inserted.

There is a known (s800) disc2 driver problem that may cause the program to hang. This driver problem will cause a non-existent Alink disk drive to be configured by autoconfig and appear to be physically present. To spot this situation, you can run ioscan using -f option and look for entries that look like the following:

Class	LU	H/W Path	Driver	H/W Status	S/W Status
-----	--	-----	-----	-----	-----
.					
.					
hpib	-	36.1	cio_ca0.hpib0	ok(0x2)	ok
hpib	-	36.2	cio_ca0.hpib0	ok(0x2)	ok
disk	8	36.3.0	cio_ca0.hpfl0.disc2	Unrecognized_HW	ok
disk	9	36.3.1	cio_ca0.hpfl0.disc2	Unrecognized_HW	ok
unknown	-	36.3.2	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
unknown	-	36.3.3	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
unknown	-	36.3.4	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
unknown	-	36.3.5	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
unknown	-	36.3.6	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
unknown	-	36.3.7	cio_ca0.hpfl0.?	Unrecognized_HW	No_Driver
.					

Notice here in the ioscan output, disc2 disk is configured but the interface card for the disk (e.g. Alink card at address 36.3) is missing, ie, contrary to what the output shows, there is no Alink card physically present at address 36.3. Consequently, the entries for addresses 36.3.0 through 36.3.7 are fictitious.

If the verifier hangs as a result of this problem, it can be killed (except for the main module, verify, which becomes a zombie), but a complete cleanup requires rebooting the system.

SYS_TEST

This is the test stimulus code used to verify the boot path on a SPU using a 68020/30/40. This code is in "a.out" format and should be booted in a standalone environment.

Similar code exists in LIF format and should be booted if a cnode is being tested in a diskless environment. Both sets of code need to exist to allow testing from both a standalone and a clustered environment.

3-86 Part 2: Release-specific Topics



To install SYS_TEST:

```
copy SYS_TEST to /
modify the boot area to include SYS_TEST
a copy of the boot are is a file named /etc/boot
```

SYS_TESTDL

This is the test stimulus code used to verify the boot path on a SPU using a 68020/30/40. This code is in LIF format and should be booted in a diskless environment.

Similar code exists in "a.out" format and should be booted on a standalone system. Both sets of code need to exist to allow testing from both a standalone and a clustered environment.

To install SYS_TESTDL:

```
copy SYS_TESTDL to /usr/boot
change /etc/boottab from:

cluster:HPS300::SYSHPUX,SYSDDEBUG,SYSBCKUP:/hp-ux,/SYSDDEBUG,/SYSBCKUP
cluster:HPS800::/usr/lib/uxbootlf:/hp-ux

to:

cluster:HPS300::SYSHPUX,SYSDDEBUG,SYSBCKUP,SYS_TESTDL:/hp-ux,
/SYSDDEBUG,/SYSBCKUP,/dev/null
cluster:HPS800::/usr/lib/uxbootlf:/hp-ux
```

_PAM (Personal Applications Manager)

PAM, (Personal Applications Manager), has been modified so that it no longer is supported as a “window smart” application. This means that PAM will behave in the X11 Windows environment in the same way it has always behaved when executed on a terminal. PAM no longer creates new windows for application programs executed from within PAM or via the VIEW key. Mouse input now has been limited so that it is now only available in the softkey area. This will be the final release for PAM (8.0 and 8.01)

_Performance and Tuning

fs_async and synchronous Disk Writes

Features

In the previous release of HP-UX, many file system data structures were written to disk synchronously. While this strategy helped insure file system integrity in the event of a crash, it was costly in terms of file system performance. Performance tests indicated that system performance could increase significantly when disk writes occurred asynchronously.

A new feature available on both the series 300 and 800 allows the user to specify whether these disk writes should be performed synchronously or asynchronously. If asynchronous writes are selected, UNIX file system semantics for NFS and diskless clustered environments are preserved. If a file is opened with the “O_SYNC” option and asynchronous writes are selected, the file will continue to be written synchronously.

Configuration

The “fs_async” parameter in the S800 gen file and the S300 dfile enables and disables the feature. A zero value specifies that the writes should be performed synchronously and a non-zero value specifies that the writes should be performed asynchronously. If this parameter is not specified, then writes are performed synchronously.

Impact

If this feature is enabled, system performance should increase for most applications. However, in the event of a system crash, file system data structures may be left in a more inconsistent state than if the feature was not enabled. Normally, recovery from a file system crash is performed automatically by “fsck” in the reboot process and does not require any system administrator intervention. However, use of this feature may require system administrator intervention in the event of a crash.

Performance data

Performance testing using a program development benchmark on the series 300 and 800 revealed that some performance gain was realized in all test cases when disk writes were done asynchronously. On the series 800, performance improvements of 6 to 26 per cent were obtained with between 3 and 30 users. The series 300 tests achieved performance improvements of 6 to 25 per cent with between 1 and 12 users.

_Powerfail Note and SCSI

Be advised that the 8.0 release of SCSI does not support system powerfail. If a SCSI device is connected to a Series 800 system, system powerfail is automatically disabled by the operating system during the startup process.

_SCSI

C1700A Optical Autochanger driver

Summary of Change

The major change is to expand support for the C1700A optical autochanger to the series 800 system. Support will continue for the series 300 but will include some enhancements made to the driver. The major enhancement for the series 300 is that an fsck of every mounted disk is not required after a power failure or a system crash.

The name of the driver has been changed on the 300 from “ac” to “autoch”. The master file has been changed to reflect that fact but any old dfiles that contain a reference to “ac” should not be used.

Configuration

Since the C1700A is a SCSI only peripheral, the autochanger driver will only be supported on a system that has a SCSI interface. The driver will be supported on both NIO and CIO cards on the 800 and on the 98658A and the 98265A SCSI cards on the 300.

Features

The user treats the autochanger as 64 mountable disks (32 cartridges, with two sided media). All 32 cartridges can be mounted into the file system and can be accessed transparently. No special program needs to be executed nor any user intervention is needed to load the proper cartridge into a drive. Implied in transparent access is the ability to access surfaces without worrying about synchronizing several users. Just as several users may access the file systems on disks so too can they access file on different cartridges. The autochanger driver takes care of this synchronization making sure that the proper media surface is in the drive.

Existing application programs do not have to change to access files on the autochanger.

Disk sections are supported on the cartridges on the series 800 only.

Impact

A new entry has been made in the `bdevsw[]` table to facilitate the marking of the surface clean before it is removed from the drive. This prevents the need to `fsck` the surface after a power failure. This should be transparent to all existing drivers including 3rd party drivers. The “`conf.c`” program created by `config` or `uxgen` will reflect this new entry point.

There are two situations that can occur which will prevent the autochanger driver from marking the surface clean. These are if there are:

1. any pipes configured on the surface, and
2. if there are any unlinked-open files on the surface.

Application programs should be careful not to create these situations on a surface. They will operate properly but if a power failure or system crash occurs, that surface must be `fsck`d. The situations arise primarily for use in temporary space allocation which for performance reasons should reside on online disk.

Performance data

Because of the mechanical nature of the device, if the file requested is on a cartridge which is not currently in a drive, a 15 second delay will occur while the cartridge is inserted. Once the cartridge is inserted, the user performance will be identical to that of the C1701A, stand-alone optical disk. Users need to be aware that when accessing files on the autochanger, some delay will occur. Treating this device as a straight winchester replacement will lead to disappointing performance.

Several performance enhancements have been made over the 7.0 driver. Since each surface has all its buffers flushed to the disk and the superbblock marked clean before it is removed from the drive and put back into a slot, the syncer will not cause excessive swapping when several cartridges are mounted read-write.

Another improvement is that after the last close of a surface, the cartridge is left in the drive for a short period of time before it is put back into its slot. If the cartridge is accessed again before this time expires, an additional swap of the cartridge is not required. This will happen when a user does a `mediainit`, `newfs`, and `mount` of the cartridge.

Size requirement

The kernel data structures needed in the driver are dynamically allocated and therefore the memory requirements are dependent upon the system load.

The size of the driver is about 5000 lines of C source with the resulting object files of about 47k on the series 300 and about 72k on the series 800.

NOTE: When configuring a system with an autochanger, the number of system buffers and inodes should be increased. Each mounted file system takes a buffer and 4k bytes of memory to hold the superblock in core. If several autochangers are connected, with each surface split into sections (on an 800), the number of buffers and inodes required becomes large.

User Limitations for S800 SCSI

Note that the hardware path to SCSI devices includes the Logical Unit (LUN) number. Currently, that number shall always be "0". Thus, a disk with its bus address (thumb-wheel switch) set to "5" which is connected to a SCSI device adapter card in slot 2 of an 835 could be denoted, "4.2.5.0" rather than "4.2.5".

_Standards and HP Conformance

For the 8.0 release, HP-UX conforms to the following standards:

- X/Open Portability Guide Issue III
- XPG3 Base Branding
- FIPS 151-1
- POSIX.1-1990 (IEEE Std 1003.1-1990)
- (also known as ISO/IEC 9945/1)
- proposed POSIX.2 FIPS (June 1990)
- ANSI C (ANS X3.159-1989)
- System V Interface Definition (SVID2)
- Base System
- All Extensions except: AS, NS, and shmat(KE_OS)
- X Window System Version 11, Release 4
- OSF/Motif 1.2

_STD/FES

New 850 PDC firmware update is required for HP-UX 8.0.

PDC firmware 3.5.6 has been released for HP9000 Model 850. This new firmware corrects several problems:

- Memory low point fix required for HP-UX 8.0 release. Starting at 8.0, kernel corruption can occur following an LPMC (single bit cache parity error) if this update is not done.
- Power Fail recovery failure (caused by multiple power fails within a 2 second window, the error message is “WARNING 35— Powerfail Recovery Aborted”).
- PDC_MODEL string does not return SW_CAP information. This will allow the 850 to use the SW_CONFIG/SW_CAP tools.

NOTE 1

Service Note 9742A-01C covers this update. Please make sure your customer base is updated prior to the release of HP-UX 8.0.

NOTE 2

PDC 3.5.6 introduced a new bug: MTEST hangs on some system configurations. MTEST is a PDC based test. This test is normally NOT used.

_Swap Space

Swapping Subsystem

Summary of Change

At 7.0, support for dynamically adding file system swap was added to the swapping subsystem. Dynamic swap allocation was completed in 8.0 by adding the functionality of allocating swap devices to a running system. The ability to assign priorities to these devices was also added. In addition, improvements were made in the allocation of swap space to resolve problems with fragmentation which led to unusable swap space.

Configuration

Two new tunable parameters “nswapdev” and “swchunk” have been added to the swapping subsystem. The parameter “nswapdev” is the number of swap devices that may be configured into this system, either statically at kernel configuration time, or dynamically at run-time. The system default for this parameter is “10”. The parameter “swchunk” is the size of each contiguous block of swap space to be allocated to the system for swapping. The default is “2 megabytes”. Additionally, the meaning of the tunable parameter “maxswapchunks” was extended to the maximum number of chunks of swap space the kernel will be able to use both in device swap and pm filesystem swap. The expression “swchunk * maxswapchunks” is the total amount of swap space that the kernel is able to use. More space may be configured into the system, but it would be wasted until the system was reconfigured and rebooted with a new value for “maxswapchunks”.

For more information on configuring your swapping subsystem please see the “System Administration” documentation.

_System Administration Manager (SAM)

New X.25 Installation Procedure

Configuration of X.25/9000 through SAM utility

SAM can now be used to install, configure, initialize, remove, and shutdown X.25/9000. SAM replaces the X25install utility. It also can be used to execute X25init and X25stop for a X.25/9000 interface that has been configured. SAM can also be used to configure the new PAD products.

SAM Backup and Recovery

In 8.0 the user is allowed to backup and recover files in SAM. This functionality was dropped from SAM in release 7.0. Users will be able to perform a backup online, recover files online, add/delete an entry in the automated backup schedule, obtain a list of files from the backup tape, and view various backup and recovery logfiles. Cartridge tapes, magnetic tapes, DAT and cartridge tape autochanger devices are supported. When performing an online backup or recovery operation, the SAM interface is suspended so that the user can read/respond to the interactive messages produced by fbackup (1M) and frecover (1M).

The script `"/usr/sam/backup/backauto.scr"` has changed for release 8.0. In order to use SAM for backup/recovery, the new `"backauto.scr"` must be used. Users who have modified `"backauto.scr"` in release 7.0 and wish to continue using their modified version should make the same modifications to the new `"backauto.scr"` script (where applicable). The old `"backauto.scr"` script will be saved in `"/tmp/backauto.scr"` upon installation of release 8.0. In addition, SAM uses a new script, `"/usr/sam/backup/recfiles.scr"`, for recovering files online. Users may customize this script as well.

Cluster Configuration with Domain Names and Name Server

A problem existed in the Cluster Configuration area of 7.0 SAM when the name server (aka named, BIND) was being used. The problem was most often seen when trying to add a client to a cluster, and the name of the client was stored by the name server as a domain name (i.e., something like “client.hp.com”). SAM would accept the hostname portion of the domain name as the client name, find and display the internet address associated with that domain name, but then produce an error message to the effect that the client name was different from the domain name (i.e., that “client” was different from “client.hp.com”).

This problem has been corrected, the Cluster Configuration functions now work properly regardless of whether hostnames and internet addresses are managed by “/etc/hosts”, the name server (aka BIND), or Yellow Pages (though SAM will only update the “/etc/hosts” file, and informs the user of the need to manually update either of the other two databases if they are being used).

SAM and Support for Designating Swap Locations

The Cluster area contains a new menu item for designating swap locations. A cnode can now be configured as a swap server, and other cnodes can swap to it instead of the root server. Cnodes are considered swap servers if they swap to themselves and have 4 “gcsp” running.

Miscellaneous Bug Fixes in SAM

The following problems in the Cluster Configuration area of 7.0 SAM have been corrected.

- When creating a cluster server, if the user elects to not have SAM reboot the system (this is the last step in converting a system to be a cluster server) then the instructions for how to do the reboot were not written the log file (even though the message displayed by “M” said that they were). This problem has been corrected as “/tmp/cluster.log” will contain the directions for properly completing the process (including reboot) converting a system to a cluster server.
- The link level address field on several of the Cluster Configuration forms allowed invalid values to be entered. The checking of values entered by the

user is now more robust, not allowing invalid characters or anything less than 12 characters.

- If an 850 or 855 machine was being converted to a cluster server, then the user had to supply a valid template file for rebuilding the kernel for the system with the diskless driver included and the diskless tunable parameters set properly (for all other systems, SAM did this automatically). This has been changed so that there are no exceptions, SAM automatically rebuilds a kernel with the appropriate drivers included and the appropriate tunable parameters set for all hardware models.
- If the file `/usr/sam/config/cnode.config` contained comment lines (i.e., a line beginning with `#`) then it was possibly for the Cluster Configuration area of 7.0 SAM to behave improperly. This problem has been corrected.
- The default S800 file for s800 clients has been renamed “HP-PB.dskless”. It still resides in `/etc/conf/gen`, and it has been made more generic so that it will work for all NIO systems.
- SAM no longer tries to validate existing cnode entries that are missing information on the “Add a Client” form. Before, if an existing cnode were missing from `/etc/hosts`, SAM would not display the internet address and refuse to add any clients until the user input a value. Now, SAM still displays the existing cnode information, but it doesn’t check for its validity when the user presses “Perform Task”.
- Users can now add a Series 400 system as a client.

SAM Kernel Configuration

Sam 8.0 will support new hardware, the new virtual memory system and autoconfiguration on the s800.

New Hardware for SAM

For 8.0, SAM supports the following systems:

- 850/855 (CIO+NIO in same SPU)
- SCSI/800
- Centronics

3-98 Part 2: Release-specific Topics

■ X.25 IP and Programmatic Access for Series 300

SAM and VM System Support

There are fewer tunable parameters required in 8.0 which enhances the ease of use in the kernel configuration areas of sam.

SAM and Series 800 Auto-Config Support

Sam now initializes itself to a more complete understanding of the current hardware, kernel configuration by utilizing the new auto-config capabilities. As part of this change, the peripheral device management functions (adding, removing or viewing the status of actual devices) found in the kernel configuration submenu have been moved to the peripheral device submenu. Users can still enter the kernel configuration submenu to add, remove or check the status of device drivers.

SAM and Peripheral Devices

Peripheral device management on the S800 SAM has been removed from the kernel configuration submenu and consolidated in the peripheral devices submenu. In 7.0 SAM, only the actions not involving any kernel changes were done under the peripheral devices submenu. This meant that a user wanting to add a new peripheral device had to enter the kernel configuration submenu to modify the kernel and then go to the peripheral devices submenu to perform whatever additional tasks necessary to make the device usable. S800 SAM now does all of the steps needed to add, remove or view peripheral devices directly from the “Peripheral Devices” submenu. This includes automatically adding device drivers when necessary. The interaction model between S800 SAM and the user has also changed as a result of using the new “ioscan”(1m) command. In 7.0, the user was required to fill in all hardware path information relating to the device in question. In 8.0, SAM uses “ioscan” to scan for the device(s) and reports the information to the user in the form of a list, with the hardware path information supplied by SAM. Thus, while a user still needs to be able to identify a peripheral device given a complete hardware path specification, the user no longer needs to manufacture a hardware path.

SAM and SCSI Disk/CD-ROM Support

SAM 8.0 will support the new SCSI disks (except for the autochanger) on the s800, and locally mounted file systems and swap servers on both the s300 and s800. It will also support CD-ROM disks and the 22xx disk series. The “Change File System Parameters” screen has been obsoleted and replaced by the “Modify a Local File System”, “Modify NFS Mount Options”, “Remove/Modify Device Swap” with the same functionality. Also, two screens have been added: “Add Device Swap” and “Add a Local File System”. The “Add Dynamic Swap” screen has been renamed “Add File System Swap”. The two “View” screens have been modified to include an additional “View Details” screen, and give much more information than previously on clusters. The “Add a Hard Disk Drive” screen on the s300 and s800 has been modified so that it allows you to choose a disk from the list of new disks instead of having to determine the hardware addresses.

SAM and Non-HP Terminal Support

SAM 8.0 will support the use of some non-HP terminals. Specifically, “vt100” and “wyse-30” terminals will be supported with the SAM message line being used to “label” softkeys where possible. If users set their TERM environment variable to:

```
vt100, vt100-am, xterm, or ansi
```

SAM will run in vt100 mode. If TERM is equal to:

```
30 or wy30
```

SAM will run in “wy30” mode. All other TERM settings will be assumed to be “hp” mode. For more information about key mappings and other details see “HP-UX System Administration Tasks” under the “sam” sub-heading.

SAM and Network Configuration

Since RFA is being obsoleted in 8.0, the configuration of RFA is being removed from the SAM menu tree.

In 8.0 SAM will allow the user to configure NFS server and client capability. This was dropped from SAM, other than the ability to mount remote file systems if NFS is already operational. SAM will provide the ability to

3-100 Part 2: Release-specific Topics

enable/disable NFS client and server capability as well as add and remove to the list of file systems mounted or exported by the client or server respectively.

A “bootp” and TFTP are new to the ARPA product in 8.0. SAM now has the ability to configure “bootp” (and TFTP implicitly when “bootp” is enabled). This allows the user to set up a “bootp” server and add “bootp” clients to be served by the bootp server. TFTP is automatically enabled if it is not already enabled.

An X.25 configuration interface will be provided by IND within the SAM configuration program; so users be able to configure the ARPA Services over X.25 via SAM.

SAM and Other Utilities

Sam 8.0 will support the invocation of customer applications from the “Other utilities” main menu item. Up to 5 levels of hierarchical customer menus are supported with optional user supplied help for any item in the hierarchy. See the associated default SAM help screens or SAM documentation in “HP-UX System Administration Tasks” for more information.

SAM and User Account UID Specification

SAM 8.0 will support user-specification of “uid” values for new and existing user accounts. To make use of this feature, enter “y” in the field labeled:

- “Modify users defaults?” (y or n)

On the “Add a New User Account” form, or in the field labeled:

- “View/Modify additional information for this user?” (y or n)

On the “View/Modify a User’s Account” form (a popup window will appear showing the current “uid” value). Changing this value will change the value of the new/modified user’s “uid”. One of the primary uses of this feature might be to coordinate “uid” values for the same user on different machines on a network. See the associated SAM help screens or the SAM documentation in “HP-UX System Administration Tasks” for more information. You may also now specify if the user may log in with the “Terminal Session Manager”. This can be specified in the same popup as the user ID on the “Add a New User Account” form.

SAM and File System Protection When Removing Users

When removing users or files from a system, there is always a possibility that the wrong user may be removed or that files belonging to a user who is removed are removed inadvertently during the removal process. Take, for instance, user “bin” who happens to “own” (from the operating system perspective) the majority of the executable commands on the system. Removing this user is obviously inappropriate. On the other hand, user “joe” who happens to “own” all of the files comprising the test suite for a project. It may be alright to remove “joe” but the test suite should be left intact and assigned to a new owner. SAM 8.0 provides two new features to help protect against inadvertent removal of users or files when removing users via SAM.

First, when prompting for the name of a user to remove from the system, SAM checks the name given against a list of names specified in the file “/usr/sam/config/rmuser.excl”. If the name matches one within the file, SAM will not remove the user. When SAM removes a user, all files (or a subset thereof) for that user are also removed, unless the ownership is to be given to another user (see the associated SAM help screens or the SAM documentation in “HP-UX System Administration Tasks” for more information).

Second, before removing a file belonging to the user, SAM checks to see if the file resides in a path which has been excluded from removal. SAM uses the file “/usr/sam/config/rmfiles.excl” to determine which paths have been excluded from removal. So, for example, if the path “/users/joe/test” is named in the file, SAM will not remove any files residing beneath that directory. SAM logs a list of all files it removes in the file “/tmp/sam_remove.log”. The files “/usr/sam/config/rmuser.excl” and “/usr/sam/config/rmfiles.excl” may be edited to contain users and directories which the administrator wishes to exclude from removal by SAM. See the associated SAM help screens or the SAM documentation in “HP-UX System Administration Tasks” for more information.

SAM and Task Customization

SAM 8.0 will support customer task customization for the tasks “Add a New User Account to the System”, “Remove a User Account from the System”, “Add a Cluster Client”, and “Remove a Cluster Client”. For each of these tasks, the customer may specify steps to be performed before SAM does its processing for the task (pre-task step) and/or steps to be performed after SAM

3-102 Part 2: Release-specific Topics

does its processing (post-task step). Pre- and post-task steps may be specified from the “Task Customization” main menu item. Each customizable task has its own set of parameters which will be passed to the customer for use in pre- and post-task processing. See the associated SAM help screens, “sam”(1M) in the HP-UX Reference manual, or the SAM documentation in “HP-UX System Administration Tasks” for more information.

You Cannot Use SAM Programmatically

It is not possible to use SAM programmatically. You can only work via the SAM user interface. For this reason it is not possible for users to have programs or scripts that make use of SAM features that are being or have been changed. The only impact HP can foresee in making changes to SAM is to hope users change possible habits that have been built around the use of SAM and to provide users with new and useful capabilities.

SAM Responds at an Interactive Speed

Sam continues to respond at an “interactive speed”. Where tasks take longer than 1 or 2 seconds a “working” message appears until the task is complete. SAM size requirements are about the same and perhaps slightly less than in 7.0 due to the use of shared libraries.

_System Verifier

The system verifier (ie, the “verify” command) is new to release 8.0. It is supported on the series 300 and the series 800. It is intended to be used by customers to verify the correct installation and/or re-configuration of a system. At present the command verifies the following devices:

HPIB	disks, mag tapes, dds tapes, cartridge tapes, and printers.
SCSI	disks, and tapes.

_Terminal I/O

New Features

Autoconfiguration

All MUX drivers on Series 800 machines support the new autoconfiguration available with the 8.0 release. Both 6-channel and 16-channel MUX cards may be freely intermixed on a CIO system, except that only the 6-channel is supported for the system console.

CIO 16-channel MUX

A 16-channel version of the MUX card is now available for Series 800 machines employing the CIO backplane. While the 6-channel CIO MUX could support input of data at 19200 Baud on no more than three channels concurrently, the 16-channel card is able to support 19200 Baud on all sixteen channels, providing that flow control has been enabled. Use of the 16-channel card also provides more terminal ports with fewer MUX cards, thus reducing the system I/O slot overhead, and improving Terminal I/O performance.

NIO 8-channel AP MUX

The MUX card supplied with all new NIO systems now supports both console and AP functionality. While the card may be used on all NIO based systems as a console, the AP functionality is available only on those systems which

support AP. This is the first support of AP functionality for any NIO based machine.

DIO 8-channel MUX

An 8-channel version of the MUX card is now available for Series 300 machines capable of supporting DIO-II cards. While otherwise compatible with the existing 4-channel MUX card, the new card will support modem connections on all eight ports.

Compatibility

Autoconfiguration

Old I/O configurations should continue to work, but it is suggested that autoconfiguration be used for best system performance. The special device files used with CIO MUX connections are different (different major and minor number values), but proper use of the autoconfiguration capabilities should result in the proper modifications being automatically made to these special files in a manner transparent to the system user. It should be noted that only the standard system defined files will be modified. Any files specially created by the user will not automatically be modified.

CIO 16-channel MUX

Some hardware settings available on the 6-channel MUX are not supported on the 16-channel MUX. Five- and six-bit data are not supported. The baud rate settings of 50, 110, 134.5, 900, 1800, 3600, and 7200 are not available.

The values for the loss-of-carrier modem timer, used with CCITT style modem connections, are different than those for the 6-channel MUX. The granularity is 25 milliseconds rather than 10 milliseconds, and the maximum value is 6375 milliseconds rather than 2550 milliseconds.

NIO 8-channel AP MUX

The new MUX card is fully backward compatible with the earlier non-AP MUX card. The AP functionality is available only when the card is connected and configured as the system console card and with the necessary AP cabling.

The card otherwise appears to be a standard MUX card. The AP feature set available is compatible with the AP feature set available on CIO based systems.

DIO 8-channel MUX

The 8-channel DIO MUX appears to the system as if it were two 4-channel MUX cards. However, while the 4-channel MUX could support modem connections on only one of its four ports, the 8-channel MUX is capable of supporting modem connections on all eight ports.

Defect Fixes

Improved Modem Processing on CIO Machines

Modem processing is much more reliable on CIO based systems now, and is more consistent with that done on NIO based systems. The SR #s which are related to the changes made for this are: 4700910307, 4700918987, 4700929398, and 5000516948.

Obsolescence

The non-AP version of the 8-channel NIO MUX will no longer be available but will continue to be supported through standard HP obsolescence cycles. Console or AP functionality can not be supported on the older card.

_Updating HP-UX

808/815 HP-UX Release 8.0 Install/Update Notes

Due to increased functionality of 8.0 and its greater use of disk space, it has become necessary to document situations that involve Series 808/815 customers who will be performing Installs or Updates to HP-UX Release 8.0.

The situations the customer and/or HP support personnel should be aware of follow:

- Updating from HP-UX Release 7.0 to 8.0 with the Update program's "match what I have" feature or MU Runtime, using a 150Mb system disk.
- Installing HP-UX Release 8.0 MU Runtime, onto a 150Mb system disk.

Given the current file and fileset mappings within the Update program, more diagnostic filesets may be selected than you actually might need or even have space for. So you will need to make some decisions depending upon which of the above situations you fall under.

- After selecting the option to update existing filesets, examine your disk space situation using the Calculate Disk Space option and check to see if any ">>" are present.
- If not enough will be left over after the update finishes, then you will need to examine the Diagnostics partition to decide if more filesets are being selected than you actually need.
 - CORE-DIAG and HPIB-DIAG for system and main peripheral support.
 - NETWRK-DIAG for networking support.
- You may wish to also de-select any other filesets which may be selected, like INSTRUMNT-DIAG, GRAPHICS-DIAG or SCSI-DIAG, that you may have no need for and would improve your disk space situation considerably. If you can't do this, then you will need to contact your HP Support Person for more detailed help.



Part 3: Removed or Obsoleted Products or Features

This part contains information about products or features that have been removed from the HP-UX system.

_AGP/DGL

AGP/DGL is being discontinued after 8.0; that is, 8.0 will be the last supported release for AGP/DGL.

All customers will be given a last change to purchase this product if they have applications that they wish to port to another API but have not yet had the opportunity to do the port.

Be assured that HP will to continue to protect customer investment by offering support of AGP/DGL on HP-UX 8.0 for 5 years.

_HP Windows/9000

A) Support:

- Support for HP Windows/9000 specifically does not include the S400 machines.
- HP Windows/9000 is being discontinued after 8.0; that is, 8.0 will be the last supported release for HP Windows/9000. All customers will be given a change to purchase the product if they have applications that they wish to port to X11 but have not yet had an opportunity to do the port.

HP will continue to protect customer investment by offering HP Windows/9000 on HP-UX 8.0 for 5 years.

_Powerfail Note and SCSI

There may be some confusion about HP's support of system powerfail with a SCSI device. Whether this was supported, or not, during the 7.0 release, it is not supported for the 8.0 release. See the same topic in Part 2 for more information.

_RFA

Remote File Access is no longer supported. You can consider it to be obsolete as of the 8.0 release of HP-UX.

_X.25/9000

X25install utility is no longer part of X.25/9000 because of the addition of X.25/9000 configuration in SAM. SAM can now be used to install, configure, and initialize X.25/9000. X25install will not work on 8.0.