International Technical Support Organization

A Holistic Approach to AIX V4.1 Migration, Volume 1 AIX, UP to SMP and Oracle

February 1996

IBM

International Technical Support Organization

Austin Center

IBM

International Technical Support Organization

A Holistic Approach to AIX V4.1 Migration, Volume 1 AIX, UP to SMP and Oracle

February 1996

> **Take Note!**
>
> Before using this information and the product it supports, be sure to read the general information under "Special Notices" on page xv.

First Edition (February 1996)

This edition applies to Version 3, Release 2, Modification Level 5 of the AIX Operating System, Program Number 5756-030, Version 4, Release 1, Modification Levels 3 and 4 of the AIX Operating System, Program Number 5765-393, and Associated IBM and Independent Software Vendor Program Products.

Order publications through your IBM representative or the IBM branch office serving your locality. Publications are not stocked at the address given below.

An ITSO Technical Bulletin Evaluation Form for reader's feedback appears facing Chapter 1. If the form has been removed, comments may be addressed to:

IBM Corporation, International Technical Support Organization
Dept. JN9B Building 45 Internal Zip 2834
11400 Burnet Road
Austin, Texas 78758-3493

When you send information to IBM, you grant IBM a non-exclusive right to use or distribute the information in any way it believes appropriate without incurring any obligation to you.

Abstract

This document takes a holistic view of the migration of a RISC System/6000 environment from AIX Version 3.2 to AIX Version 4.1. It covers the migration of complete systems—systems that include not only the AIX operating system, but also applications, networking customizations, user information and the many other factors that make each installation unique.

This volume deals specifically with the migration of the AIX operating system, licensing and packaging issues, migration from a uni-processor system to one using symetric multi-processing, and the migration of systems using the Oracle Database. It also includes a detailed description of the migration process.

This document is written for current users of AIX Version 3.2.5 and associated products. Basic knowledge of the AIX operating system and the specific environment to be migrated is assumed.

(330 pages)

Contents

Figures

Tables

Special Notices

This publication is intended to help administrators of systems that are currently running AIX Version 3.2 to migrate their systems to AIX Version 4.1. The information in this publication is not intended as the specification of any programming interfaces that are provided by AIX. See the PUBLICATIONS section of the IBM Programming Announcement for AIX for more information about what publications are considered to be product documentation.

References in this publication to IBM products, programs or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM product, program, or service is not intended to state or imply that only IBM's product, program, or service may be used. Any functionally equivalent program that does not infringe any of IBM's intellectual property rights may be used instead of the IBM product, program or service.

Information in this book was developed in conjunction with use of the equipment specified, and is limited in application to those specific hardware and software products and levels.

IBM may have patents or pending patent applications covering subject matter in this document. The furnishing of this document does not give you any license to these patents. You can send license inquiries, in writing, to the IBM Director of Licensing, IBM Corporation, 500 Columbus Avenue, Thornwood, NY 10594 USA.

The information contained in this document has not been submitted to any formal IBM test and is distributed AS IS. The information about non-IBM (VENDOR) products in this manual has been supplied by the vendor and IBM assumes no responsibility for its accuracy or completeness. The use of this information or the implementation of any of these techniques is a customer responsibility and depends on the customer's ability to evaluate and integrate them into the customer's operational environment. While each item may have been reviewed by IBM for accuracy in a specific situation, there is no guarantee that the same or similar results will be obtained elsewhere. Customers attempting to adapt these techniques to their own environments do so at their own risk.

Any performance data contained in this document was determined in a controlled environment, and therefore, the results that may be obtained in other operating environments may vary significantly. Users of this document should verify the applicable data for their specific environment.

The following document contains examples of data and reports used in daily business operations. To illustrate them as completely as possible, the examples contain the names of individuals, companies, brands, and products. All of these names are fictitious and any similarity to the names and addresses used by an actual business enterprise is entirely coincidental.

Reference to PTF numbers that have not been released through the normal distribution process does not imply general availability. The purpose of including these reference numbers is to alert IBM customers to specific information relative to the implementation of the PTF when it becomes available to each customer according to the normal IBM PTF distribution process.

The following terms are trademarks of the International Business Machines Corporation in the United States and/or other countries:

AIX	AIXwindows
C Set ++	ESCON
GXT1000	GXT150L
GXT150M	IBM
InfoExplorer	Micro Channel
OS/2	Portmaster
POWER Gt3	POWER Gt3i
POWER Gt4	POWER Gt4e
POWER Gt4i	POWER Gt4x
POWER Gt4xi	PowerPC
POWERserver	POWERstation
RISC System/6000	S/370
S/390	SP1
SP2	System/370
Ultimedia	Xstation Manager
400	

The following terms are trademarks of other companies:

C-bus is a trademark of Corollary, Inc.

Windows is a trademark of Microsoft Corporation.

PC Direct is a trademark of Ziff Communications Company and is used by IBM Corporation under license.

UNIX is a registered trademark in the United States and other countries licensed exclusively through X/Open Company Limited.

DCE, Motif	The Open Software Foundation
DECstation	Digital Equipment Corporation
HP	Hewlett-Packard
INed	INTERACTIVE Systems Corporation
IPX, NetWare	Novell, Inc.
NCK, NCS	Apollo Computer, Inc.
Network Computing Kernel	Appollo Computer, Inc.
NeXT	NeXT Computer, Inc.
NFS	Sun Microsystems, Inc.
NT	Microsoft
OpenGL	Silicon Graphics, Inc.
Oracle	Oracle Corporation
PCMCIA	Personal Computer Memory Card International Association
POSIX	Institute of Electical and Electronic Engineers

SCSI	Security Control Systems, Inc.
X/Open	X/Open Company Limited
X Windows	Massachusetts Institute of Technology

Other trademarks are trademarks of their respective companies.

Preface

The Webster's Ninth New Collegiate Dictionary defines holistic as:

1: of or relating to holism 2: relating to or concerned with wholes or with complete systems rather than with the analysis of, treatment of, or dissection into parts.

This second definition describes the approach we have taken with the migration of systems from AIX Version 3.2 to Version 4.1. In the real world, a migration doesn't involve just the operating system. A computing environment includes applications from both IBM and independent software vendors and a great deal of site-specific customization of system and user environments. In this case, there are always going to be unique migration actions that cannot be performed automatically by migration tools.

Our testing included systems running applications such as HACMP and the Oracle database. It also included systems and groups of systems with extensive customizations in TCP/IP and SNA communications protocols. While it is obviously not possible to cover every specific environment, we hope that the environments we have chosen will provide a wide enough cross section to give system administrators clues or hints as to the problems they may discover in their own environments.

This document is the second part of a three-volume series on migration. The contents of each volume are listed below:

- *A Holistic Approach to AIX V4.1 Migration, Planning Guide*

  This covers the advance planning required for a successful migration. It includes such topics as documenting the existing system, determining the required levels of software for the new environment, verifying that hardware is supported at the new level, and scheduling the migration.

- *A Holistic Approach to AIX V4.1 Migration, Volume 1*

  This is the first of two volumes to assist system administrators in actually performing the migration. It includes information on the base operating system, migrating from Uni-Processor systems to those using Symetric Multi-Processors, and the migration of systems using the Oracle database. It also includes a detailed description of the migration process.

We expect to publish the following redbook in the near future.

- *A Holistic Approach to AIX V4.1 Migration, Volume 2*

  The final volume covers communications issues, such the migration of systems with complex network configurations using TCP/IP and SNA, as well as the migration from an X Windows and Motif window-manager environment to the Common Desktop Environment. It also includes sections on the migration of systems running HACMP, and the migration of large numbers of systems in various environments.

How This Document is Organized

The document is organized as follows:

- Chapter 1, "Migrating AIX from V3.2.5 to V4.1"

  This chapter covers the migration of AIX V3.2 to AIX V4.1 on a stand-alone system. The migration of the AIX operating system is one of the necessary steps in the migration of a whole customer environment.

- Chapter 2, "Installing iFOR/LS"

  This chapter helps in understanding and installing the iFOR/LS licensing system that is necessary to handle AIX V4.1 licensed products after the migration.

- Chapter 3, "How it Works"

  This chapter describes in detail the flow of a migration installation from the initial system boot, through the migration of the Base Operating System (BOS) and Licensed Porgram Products (LPPs), to the first system reboot.

- Chapter 4, "UP to SMP Migration"

  This chapter describes how to upgrade a Uniprocessor system to a Multiprocessor system. It explains how to actually perform the overall upgrade with a specific focus on how to enable the MP kernel to take advantage of the SMP, and how to reconfigure asynchronous devices.

- Chapter 5, "Migrating from AIX V3.2.5 to AIX V4.1 with Oracle"

  This chapter covers the migration of an Oracle environment from AIX V3.2.5 to AIX V4.1. It gives the system administrator guidelines and hints and tips on how to successfully achieve the migration of an entire Oracle environment to AIX V4.1.

- Appendix A, "AIX V4.1 Packaging"

  This appendix describes how AIX V4.1 is packaged.

- Appendix B, "AIX V4.1 Software Maintenance"

  This appendix describes the AIX V4.1 software maintenance mechanism called VRMF.

- Appendix C, "AIX V4.1 Compatibility Filesets"

  This appendix provides the content of the main compatibility filesets included in AIX V4.1.

- Appendix D, "rmlpp Shell Script"

  This appendix provides the content of a script shell that allows the system administrator to remove, if necessary, products on an AIX V3.2 system prior to the migration.

- Appendix E, "AIX V3.2 to AIX V4.1 Fileset Mapping"

  This appendix gives the fileset mapping used between AIX V3.2 and AIX V4.1 during the BOS migration.

- Appendix F, "Device Configuration Recovery Files"

  This appendix provides the content of some shell scripts and configuration files useful to recover device configurations.

- Appendix G, "AIX V4.1 Device Driver Packages"

  This appendix gives an AIX 4.1.4 list of device drivers packages and their description. It is intended to help the system administrator in retrieving the name of a device driver package required to support a specific hardware.

Related Publications

The publications listed in this section are considered particularly suitable for a more detailed discussion of the topics covered in this document.

- *AIX Version 4.1 Getting Started*, SC23-2527
- *AIX Version 4.1 Installation Guide*, SC23-2550
- *Common Diagnostics and Service Guide*, SA23-2687
- *Service Request Number Cross-Reference*, SA23-2629
- *AIX Version 3.2 Commands Reference*, GBOF-1802
- *AIX Version 3.2 System Management Guide: Operating System and Devices*, GC23-2486
- *AIX Version 4.1 Commands Reference*, SBOF-1851
- *AIX Version 4.1 System Management Guide: Operating System and Devices*, SC23-2525
- *AIX Version 4.1 Network Installation Management Guide and Reference*, SC23-2627
- *AIX Version 4.1 Files Reference*, SC23-2512
- *All About AIX Version 4.1*

  This book was never officially published in hardcopy; however, it is available on the World-Wide Web at the URL:

  `http://www.developer.ibm.com/sdp/library/ref/about4.1/df4main.html`

  Within IBM, *All About AIX Version 4.1* is also available as ABOUT4_1 PACKAGE on the MKTTOOLS disk.

International Technical Support Organization Publications

A complete list of International Technical Support Organization redbooks, with a brief description of each, may be found in:

To get a catalog of ITSO redbooks online, VNET users may type:

`TOOLS SENDTO WTSCPOK TOOLS REDBOOKS GET REDBOOKS CATALOG`

A listing of all redbooks, sorted by category, may also be found on MKTTOOLS as ITSOCAT TXT. This package is updated monthly.

ITSO Redbooks on the World Wide Web (WWW)

Internet users may find information about redbooks on the ITSO World Wide Web home page. To access the ITSO Web pages, point your Web browser to the following URL:

`http://www.redbooks.ibm.com/redbooks`

IBM employees may access List3820s of redbooks as well. The internal Redbooks home page may be found at the following URL:

`http://w3.itsc.pok.ibm.com/redbooks/redbooks.html`

Acknowledgments

This project was designed and managed by:

Yves Bex
Cameron Ferstat
International Technical Support Organization, Austin Center

The authors of this document are:

Kriangsak Chanpiwat
IBM Thailand

Colin Fearnley
IBM South Africa

Ewa Frisch
IBM Austria

This publication is the result of a residency conducted at the International Technical Support Organization, Austin Center.

Chapter 1. Migrating AIX from V3.2.5 to V4.1

This chapter covers the AIX migration from Version 3.2.5 to Version 4.1 on a stand-alone system. The objective of this chapter is to help you prepare and perform the migration of AIX. This migration is one of the necessary steps in the migration of an entire customer environment from AIX Version 3.2 to AIX Version 4.1.

This chapter applies to both AIX V4.1.3 and AIX V4.1.4. Differences in the behavior of these two modification levels of AIX V4.1 will be highlighted.

Networking aspects are not covered in this chapter and will be covered in *A Holistic Approach to AIX V4.1 Migration, Volume 2*, SG24-4653.

1.1 AIX V4.1 Installation Methods

This following section is a review of the AIX V4.1 installation methods. It is intended to help you in choosing the most appropriate method to upgrade your system. Note that we will focus in the rest of the document on the Migration Install method.

There are four (4) methods that can be used to upgrade AIX Version 3.2 to AIX Version 4.1. These methods are:

- New or Complete Overwrite Install
- Preservation Install
- Migration Install
- mksysb Install

Note: One can use any of the above methods to install a system from the network. Even though installing a system through a network is sometimes referred to in this document as a Network Install, it is not a separate installation method but an installation environment using Network Installation Management (NIM). Also, when migrating from AIX V3.2.5 to AIX V4.1, the migration install method is the *default* method.

1.1.1 New or Complete Overwrite Install

This method should be used to install AIX V4.1 on a new system or on a system on which you want to overwrite the existing version of AIX.

The main advantage of this method is that your resulting AIX V4.1 installation is perfectly clean. On the other hand, the entire system will need to be reconfigured. The amount of work depends, of course, on the complexity of your installation. This work must be evaluated before choosing such a method.

User-defined volume groups (non-root volume groups) will be preserved and will remain there after the new installation. You will just have to import these volume groups, and mount the file systems created on them.

Following are some of the tasks you might have to perform. These tasks depend on your specific environment.

- Set the system date and time for your time zone.
- Set a root user account password to restrict access to system resources.
- Confirm or change the install device you want to use for installing additional software. The device may be a CD-ROM, a tape drive or a local or remote directory.
- Check the system storage and paging space needed for installing and using additional software applications.
- Set your National Language Support (NLS) environment.
- Import user-defined volume groups.
- Create user accounts and passwords.
- Install third-party device drivers (for example, LAN printer server)
- Set your system network configuration (if applicable).
- Create local and remote terminals.
- Configure local and remote printers.
- Install and configure additional LPPs.
- Install and configure third-party software.

This method can be suitable for a stand-alone system with a simple configuration.

1.1.2 Preservation Install

Preservation Install is the default installation method of the Base Operating System (BOS), for systems running AIX Version 3.1 or AIX Version 4.1. For systems running AIX Version 3.2, the Migration Install is the default method.

With the Preservation Install method, the contents of /usr, /, /var, and /tmp will be deleted.

The Preservation Install will save the previous paging space and dump device, /home and other user-created file systems in rootvg.

User-defined volume groups (non-root volume groups) will be preserved and activated automatically after the Preservation Install. Before mounting the file systems belonging to these user-defined volume groups, you might need to recreate the file systems' mounting directories that were removed during the installation (directories under / or /usr for example).

Configuration files, with the exception of /etc/filesystems, will be deleted. If you want any additional configuration files to be saved during the Preservation Install, you must edit the /etc/preserve.list file on your AIX V3.1 system, and add the full path names of the configuration files you want to save. The system needs sufficient disk space in the /tmp file system to store the files listed in the /etc/preserve.list file.

This installation method should be used when the system is running AIX V3.1, and you want to preserve the user data in the rootvg volume group.

1.1.3 Migration Install

This installation method is the default option for systems running AIX Version 3.2.X. A Migration Install allows you to upgrade your system from AIX Version 3.2 to AIX Version 4.1, while keeping customized configuration information and installed, optional software.

Any configuration files that cannot be migrated will be saved in a specific directory under /tmp. System messages will inform you of the location of the saved files. Information will also be stored in the system installation log in /var/adm/ras/devinst.log.

The Migration Install will preserve all file systems including /, /usr and /var.

If the AIX V3.2 system has /dev/hd7 used as the dump device, this dump device will be removed during the migration. After the migration, the paging space (/dev/hd6) will be used as the dump device. This is a way to gain some extra space for the migration.

Note: When used as the dump device, /dev/hd7 should only be removed when it uses two partitions. In AIX V4.1.3 and AIX V4.1.4, it is actually removed whatever its size during the migration.

If a specific dump device (a specific logical volume) has been created prior to the migration, it will be preserved.

All files in /usr/lib/drivers, /usr/lib/microcode, /usr/lib/methods, and /dev will be removed from the system.

Attention!

Non-AIX device drivers will be deleted during the migration and must be reinstalled. Non-AIX software products will remain on the system. Migration Install migrates only the products that do not run on AIX Version 4.1. Any product that was installed on the AIX Version 3.2 system, and can run on an AIX Version 4.1 system, will not be altered during the Migration Install.

The following filesets will be removed during the migration:

```
aic.obj
aic12.obj
aicm.*.msg
aic12m.*.msg
bosadt.xde.obj
bosext1.ecs.obj
bosext1.vdidd.obj
bosext2.lrn.info
bsm.*.msg
gPfonts.*
OpenGL.*
pcsim
pcsim.obj
pcsimm.*.msg
softPEX.obj
softgP.obj
txtfmt.graf.obj
txtfmtkj.*
vdigft.obj
vdiplot.obj
xlccmp.obj
X11_3d.*
X11_3diEn_US.info
```

In addition, `bosext2.dlc.*` will be removed, except for those datalink controllers which were actually configured. Also `xlfcmp.obj` and `xlfcmpm.*.msg` will be removed if installed at a version less than Version 3.

Note: If a new version of a software product is installed during the Migration Install, user configuration files from the previous version of this product will be preserved.

1.1.4 mksysb Install

A system can also be installed with AIX Version 4.1 by restoring a backup of an other AIX V4.1 system. Since an AIX system backup is made with the `mksysb` command, this installation method will be referred to in the rest of the document as a mksysb Install.

This method can be used to upgrade (actually install) a system that have a very similar configuration to the original system. Using it for installing a system that has a very different configuration has some limitations. In any case, specific information system, such as groups and users, hostname and so on, will still need to be reconfigured.

The mksysb utility has been enhanced in AIX V4.1 to facilitate backup/restore operations, but has some limitations. For enhancements and limitations of the AIX V4.1 system backups, refer to 1.6, "Backing Up your AIX V4.1 System" on page 57 in this chapter, and 1.3.1, "mksysb Limitations in AIX V3.2.5" on page 30.

1.1.5 Installing via a Network

All the installation types discussed before (New Install, Preservation Install, Migration Install, and mksysb Install) can be also done through a network by using Network Installation Manager (NIM). The difference here is that a NIM server is used instead of an installation device (CD-ROM or tape drive).

In order to perform a Migration Install via a Local Area Network (LAN), NIM must be installed and configured on a server running AIX V4.1. The installation resources, a spot and the AIX Version 3.2.5 systems to be migrated, will have to be defined as objects on the NIM server.

Installation resources have to be allocated to the system objects, and then the chosen installation method can be performed from the NIM server.

A Migration Install using NIM is suitable for migrating multiple LAN-connected systems. You will find more information on how to migrate multiple systems using NIM in the following redbook: *A Holistic Approach to AIX V4.1 Migration, Volume 2*, SG24-4653.

1.2 Preparing the Migration

We assume in the rest of this document that you have carefully read the redbook *A Holistic Approach to AIX V4.1 Migration, Planning Guide*, SG24-4651. At this step, you must have checked if all the applications, License Program Products (LPPs) and device drivers you need are available on AIX V4.1.

The objective of this section is to help you in preparing the AIX migration. There are several things you need to do, or check, before starting the migration. The following list is not exhaustive.

- Get the relevant documentation to perform the migration.
- Check the level of your system.
- Check your installed LPPs.
- Test your custom applications in AIX V4.1, and port them if necessary.
- Test your custom shell scripts in AIX V4.1, and port them if necessary.
- Choose the appropriate method for upgrading your system.
- Check your users and groups definitions.
- Check your scheduled jobs, reschedule them and save them.
- Check your non-rootvg volume groups.
- Check your disk space, and if necessary, get some extra disk space.
- Document your system.
- Check the root user primary authentication method.
- Estimate the migration time.
- Prepare recovery procedures.
- Back up your entire system.

1.2.1 Checking Your Operating System Level

We recommend that you first check which level of AIX you are currently running.

If your system is running AIX V3.2.4 or AIX V3.2.5, you can use the `oslevel` command. This command takes some time to send you a final output. Following is an example of the command output on an AIX V3.2.5 system:

```
# oslevel
Processing ... Please Wait
>3250
```

For systems running AIX V3.2.0 up to AIX V3.2.3, it is very difficult to know exactly which AIX level you are running. If the `oslevel` command is not available on your system, run the `lslpp -l` command.

The Migration Install is supported for every level of AIX V3.2. Thus it is possible to migrate an earlier AIX V3.2.X system to AIX V4.1. However, in this document we only tested the migration from AIX V3.2.5 to AIX V4.1.

Upgrading an earlier AIX V3.2.X system to AIX V3.2.5 is possible and should not cause major changes to AIX. Most applications should be able to run unchanged after the upgrade. BUT going from AIX V3.2.X to AIX V3.2.5 will take as much or more time than migrating directly from AIX V3.2.X to AIX V4.1.

In the rest of the document, we assume that you are migrating from AIX V3.2.5.

1.2.2 Porting In-House Applications

It is not our intent in this chapter to cover application porting issues to AIX V4. We assume that third-party applications have been ported prior to the migration. In-house applications must also be tested and/or ported. Many porting issues are already covered in the *All About AIX Version 4*. We strongly recommend that you refer to this document.

1.2.3 Testing Custom Shell Scripts

You might have specific shell scripts written in C-Shell, Bourne Shell or Korn Shell on your system. If possible, these procedure should be tested on an AIX V4.1 testing system prior to the migration.

In order to meet compliance to standards, such as POSIX 1003.2 and XPG4, some commands have changed; some of them have been removed. These shell scripts may use some non-standard commands that have been removed, such as the `del` command or the `copy` command, for example. Also, these shell scripts may use user commands whose options have slightly changed to be standards compliant. New options should not affect the behavior of a shell script, but options that are no longer supported or that have a different output should be checked.

1.2.3.1 Removed Commands

Most of the removed commands are, in fact, still shipped within the bos.compat.cmds compatibility fileset which is automatically installed during a Migration Install. Refer to the section 1.2.4, "Understanding Compatibility Filesets" on page 10.

1.2.3.2 User Command Changes

This is a list of commands whose options have changed slightly:

- acctdusg
- admin
- axeb
- awk
- backbyname
- bc
- bosboot
- bsh
- catman
- chuser
- cksum
- cmp
- cp
- cpio
- csh
- ctags
- date
- delta
- df
- diff
- diskusg
- du
- ebxa
- echo
- ed
- expr
- fold
- fsdb
- getconf
- head
- iconv
- istat
- join
- ksh

- ksh built-in commands
  - echo
  - export
  - fc
  - hash
  - jobs
  - kill
  - readonly
  - set
  - trap
  - umask
  - unalias
  - unset
  - wait
- lex
- locale
- localedef
- lp
- lsfs
- lsjfs
- man
- more
- nice
- nm
- nohup
- od
- pack
- paste
- pax
- renice
- sed
- sort
- strings
- stty
- tctl
- tee
- touch
- uniq
- wc
- what
- yacc

If you use one of these commands within your shell scripts, we recommend that you look at the description of the command changes in the *All About AIX V4.1* document.

1.2.3.3 LFT Non-Supported Commands

Since the HFT (High Function Terminal) has been replaced by the LFT (Low Function Terminal), all commands specific to the HFT are no longer supported. This can have an impact on your existing shell scripts. The open command is a good example. Many users used to open a new virtual terminal before starting an application. This is not supported in AIX V4.1.

Following is a list of commands that are not supported by the LFT.

- chcolor
- chcursor
- chhwkbd
- chkeymap
- chnumvt
- chsound
- gm
- lscolor
- lsscreen
- mkkbd
- open
- swkbd

Note also that some commands are still supported on the LFT, but their behavior or their format have slightly changed.

- chdisp: improved format

- chfont: does not take effect until next IPL

- chkbd: does not take effect until next IPL

- lsdisp: improved format

- lsfont: improved format

- lskbd: improved format

- mkfont: does not take effect until next IPL

All these changes can have an impact on your custom shell scripts.

1.2.3.4 Printer Name Changes

You should be aware that printer names are different in AIX V4.1. For example, the 4019 printer is called 4019 in AIX V3.2.5 and ibm4019 in AIX V4.1. This might cause a problem when running a shell script that calls the `mkvirprt` command. You will find a list of printer names that have changed in the *All About AIX Version 4* document.

1.2.4 Understanding Compatibility Filesets

In order to provide binary compatibility between AIX V3.2 and AIX V4.1, compatibility filesets are installed during the Migration Install. The compatibility filesets that are actually installed depend on the LPPs or Optional Program Products (OPPs) that were previously installed on your AIX V3.2.5 system. Appendix E, "AIX V3.2 to AIX V4.1 Fileset Mapping" on page 311 shows the fileset mapping that occurs when migrating from AIX V3.2 and AIX V4.1.

These filesets offer commands, library versions, symbolic links, and other functions that make AIX Version 4.1 look more like AIX Version 3.2.5 from the application point of view. For example, removed commands are still shipped within AIX Version 4.1 within the bos.compat.cmds fileset.

You might need to use these filesets after the migration to run your shell scripts unchanged, or you might want to change your shell scripts in order to be more standards compliant.

Note: These compatibility filesets might not be provided in future AIX releases.

The compatibility filesets have their name characterized by the string `compat` in their name. The filesets are organized as follows:

- Base operating system commands
- Base operating system libraries
- Base operating system curses/termcap
- Base operating system network installation tools
- Base operating system directories/files symbolic links
- Messages
- X11R3 libraries
- X11R4 libraries
- X11 fonts
- Motif 1.0 libraries
- Motif 1.1.4 libraries

Following is a table with the AIX Version 4.1 filesets that contain the string `compat` as a part of the fileset name.

Table 1 (Page 1 of 2). Compatibility Filesets in AIX Version 4.1.	
Fileset Name	**Description**
bos.compat.cmds	AIX Version 4.1 Compatibility Commands
bos.compat.imk	Base System Locale AIX Version 3.1 Compatibility Links
bos.compat.lan	LAN COMIO Compatibility Software
bos.compat.libs	AIX Version 3.2 Compatibility Libraries
bos.compat.links	AIX Version 3.2 to AIX Version 4.1 Compatibility Links
bos.compat.msg	AIX Version 3.1 Message Compatibility Links
bos.compat.net	AIX Version 3.2 TCP/IP Compatibility Commands
bos.compat.NetInstl	AIX Version 3.2 Network Installation Tools
bos.compat.termcap	AIX Version 3.2 Termcap Source and Library
bos.compat.termcap.data	AIX Version 3.2 Termcap Source Data
bos.loc.pc_compat.com	Common Locale Support - PC Code Set
bos.loc.pc_compat.fnt	Base System Locale PC Compatibility Fonts
bos.loc.pc_compat.[lang]	Base System Locale PC Code Set - [lang]
bos.msg.en_US.compat.cmds	AIX Version 4.1 Compatibility Command Messages - U.S. English
X11.compat.fnt.pc	AIXwindows PC850 Fonts Compatibility
X11.compat.fnt.oldX	AIXwindows Ancient X Consortium Fonts Compatibility
X11.compat.lib.Motif10	AIXwindows Motif 1.0 Libraries Compatibility

Table 1 (Page 2 of 2). Compatibility Filesets in AIX Version 4.1.	
Fileset Name	**Description**
X11.compat.lib.Motif114	AIXwindows Motif 1.1.4 Libraries Compatibility
X11.compat.lib.X11R3	AIXwindows X11R3 Libraries Compatibility
X11.compat.lib.X11R4	AIXwindows X11R4 Libraries Compatibility
X11.compat.samples.util	AIXwindows Sample Utilities Compatibility
X11.x_st_mgr.compat	Xstation Manager Compatibility

You will find below a short description of the major compatibility filesets:

bos.compat.cmds This fileset contains commands that are being removed from the AIX system. These commands are generally not standards compliant. For example, the `copy` and `li` commands are part of this fileset.

bos.compat.imk This fileset provides input method keymaps and links which may be necessary for those applications that depend on the input method conventions used in AIX Version 3.1.

bos.compat.links This fileset provides backwards compatility for those links that were added in the AIX Version 3.2 release for compatibility with AIX V3.1, but have been removed in the AIX Version 4.1 release. Note that these links might not be provided in future AIX releases.

bos.compat.lan Provides COMIO interfaces for token-ring, Ethernet and FDDI devices that are compatible with the COMIO interface present in AIX Version 3.2.5. It supports usermode access only.

bos.compat.msg This fileset provides symbolic links for backwards compatibility with message catalog packages that use the AIX Version 3.1 naming conventions for language-territory.

bos.compat.net This fileset provides backwards compatibility for those links that were added in the AIX Version 3.2 bosnet.obj product.

bos.compat.NetInstl This fileset provides the capability to serve AIX V3.2 installation images to AIX V3.2 clients from an AIX V4.1 server.

bos.compat.termcap This fileset provides an older method of defining terminal definitions. Most people use terminfo. The library provides terminfo interface routines for curses applications. The same routines are provided by libcurses.a, as well.

bos.compat.termcap.data This fileset provides the termcap tabset files for various terminals.

There are also a number of compatibility filesets for X11R3, X11R4, Motif 1.0, Motif 1.1.4, and X11 fonts. If an application is not working on AIX V4.1, check the list of installed compatibility filesets.

The following example shows the content of a typical compatibility fileset. The `bos.compat.cmds` fileset is used in this example.

```
#lslpp -f bos.compat.cmds | pg
  Fileset            File
  -------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.cmds 4.1.3.0
                     /usr/lib/security/scan/xa1clean.lst
                     /usr/lib/security/scan/virsig.lst
                     /usr/bin/msgs
                     /usr/bin/del
                     /usr/bin/page -> /usr/bin/more
                     /usr/bin/copy -> /usr/bin/cp
                     /usr/bin/virscan
                     /usr/lib/security/scan
                     /usr/lib/security/scan/ghoclean.lst
                     /usr/bin/move -> /usr/bin/mv
                     /usr/bin/joinconf -> /usr/bin/confer
                     /usr/bin/confer
                     /usr/lib/security/scan/648clean.lst
                     /usr/lib/security/scan/darclean.lst
                     /usr/ucb/more -> /usr/bin/more
                     /usr/lib/security/scan/vacclean.lst
                     /usr/bin/di -> /usr/bin/li
                     /usr/bin/li
                     /usr/bin/delete -> /usr/bin/rm
                     /usr/ucb/sccs -> /usr/bin/sccs
```

For your convenience, in Appendix C, "AIX V4.1 Compatibility Filesets" on page 295, you will find the content of each compatibility fileset.

1.2.5 Checking Users and Groups Definition

For the migration process to function correctly, system users and groups definitions in the /etc/group and /etc/passwd files must be verified.

Following is a list of entries that should exist in the /etc/passwd and /etc/group before the migration to AIX Version 4.1. If an entry is missing or different from the one shown below, you might encounter problems after the migration.

Therefore, make sure that the system user and group definitions match the entries in the following list.

- /etc/passwd file

  ```
  root:!:0:0::/:/bin/ksh
  deamon:!:1:1::/etc:
  bin:!:2:2::/bin:
  sys:!:3:3::/usr/sys:
  adm:!:4:4::/var/adm:
  uucp:!:5:5::/usr/lib/uucp:
  guest:!:100:100::/home/guest:
  nobody:!:4294967294:4294967294::/:
  lpd:!:9:4294967294::/:
  ```

- /etc/group file

```
system:!:0:root
staff:!:1:
bin:!:2:root,bin
sys:!:3:root,bin,sys
adm:!:4:bin,adm
uucp:!:5:uucp
mail:!:6:
security:!:7:root
cron:!:8:root
printq:!:9:
audit:!:10:root
ecs:!:28:
nobody:!:4294967294:nobody,lpd
usr:!:100:guest
```

There is a possibility that if an entry is missing or different, the migration might be unsuccessful, or the user definitions could become corrupted.

We then suggest that you use the following commands to check if your passwd and group files do not have any anomalies.

The following command will verify the correctness of all user definitions in the user database. It will send you the result, but will not correct any errors.

```
# usrck -n ALL
```

The following command will verify the correctness of all groups, but will not correct any errors.

```
# grpck -n ALL
```

So that you can control any needed modifications, we do not recommend that you ask for an automatic error correction using the -y option.

1.2.6 Checking Scheduled Jobs

If the system you are migrating is a server, it probably has scheduled jobs to be executed. The cron daemon usually runs shell commands or shell scripts at specified dates and times. The cron daemon is started automatically out of the /etc/inittab file and respawns if it dies. It checks the directories /var/spool/cron/crontabs and /var/spool/cron/atjobs and executes the jobs defined there.

Jobs submitted by root or any user with the at command are temporarily stored in /var/spool/cron/atjobs as files with names like root.817794000.e or username.817794000.e.

To check if you have jobs that have been scheduled with the at command, enter the following:

```
# at -l
```

> **Attention!**
>
> We noticed on some systems that after the migration, jobs scheduled with at were no longer pending and were executed during the migration.

In fact, on some systems, the system date is set during the migration to a value greater then the job scheduled dates. Thus the jobs are executed during the migration. Since this problem does not show up on all systems, you might need to reset the system date to the current date after the migration.

If `at` jobs are scheduled, we recommend the following:

- Reschedule jobs that are scheduled during the migration time.

- Save the content of /var/spool/cron/atjobs on a diskette.

- Remove the content of /var/spool/cron/atjobs to avoid jobs being executed during the migration.

- Check again with `at -l` that jobs are no longer scheduled.

We also recommend that you save on the same diskette the following files that you might have created or customized:

- /usr/adm/cron/at.allow
- /usr/adm/cron/at.deny
- /usr/adm/cron/cron.allow
- /usr/adm/cron/cron.deny

We noticed that after the migration, at.allow and cron.allow were preserved, but even if at.deny and cron.deny were still there, their content was lost (all the user names defined in this file were lost). You normally can retrieve these two files from /tmp/bos/var/adm/cron after the migration, but in our tests, we only found the at.deny file and not the cron.deny file.

We also noticed that the crontab files for users like root, sys and adm are set to a default AIX V4.1 value. This means that if you customized, for example, the crontab file for root, this file will be saved in /tmp/bos/var/spool/cron/crontabs after the migration. It is then possible after the migration to retrieve the lines you added or modified in the root crontab file, and insert them in the new root crontab file. In fact, it was deemed too dangerous to automatically merge the crontab files from AIX V3.2. These files will need to be merged manually.

For example, if you uncommented the /etc/skulker entry and added a job to be executed every day at 3:00 a.m., before the migration you will see:

```
# crontab -l
# @(#)08 1.15  com/cmd/cntl/cron/root, bos, bos320 9/9/91 06:04:47
#
# COMPONENT_NAME: (CMDCNTL) commands needed for basic system needs
#
# FUNCTIONS:
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989,1991
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
0 3 * * * /etc/skulker
#45 2 * * 0 /usr/lib/spell/compress
#45 23 * * * ulimit 5000; /usr/lib/smdemon.cleanu > /dev/null
0 11 * * * /usr/bin/errclear -d S,O 30
0 12 * * * /usr/bin/errclear -d H 90
01 4 * * * /etc/lpp/diagnostics/bin/test_batt 1>/dev/null 2>/dev/null
01 3 * * * /etc/lpp/diagnostics/bin/run_ela 1>/dev/null 2>/dev/null
0 3 * * * /job
```

After the migration, the crontab file for root will be:

```
#crontab -l
# @(#)08 1.15.1.3  src/bos/usr/sbin/cron/root, cmdcntl, bos411, 9428A410j
#
# COMPONENT_NAME: (CMDCNTL) commands needed for basic system needs
#
# FUNCTIONS:
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989,1994
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#0 3 * * * /usr/sbin/skulker
#45 2 * * 0 /usr/lib/spell/compress
#45 23 * * * ulimit 5000; /usr/lib/smdemon.cleanu > /dev/null
0 11 * * * /usr/bin/errclear -d S,O 30
0 12 * * * /usr/bin/errclear -d H 90
```

You will then need to retrieve the content of the previous file in
/tmp/bos/var/spool/cron/crontabs/root file for user root.

On the other hand, user crontab files are preserved and remain there after the
migration. For example, if user yves created his own crontab file using the crontab
-e command, this file, present in /usr/spool/cron/crontabs called yves, will be
preserved.

Note: The /usr/spool/cron/crontabs directory holds all users' crontab files.

1.2.7 Access Control Lists

This facility included in AIX V3.2.5 can provide much finer-grained access control to resources (files) than the permission bits. You might use Access Control Lists (ACLs) in your system.

We set up ACLs on our testing system in order to check if they were preserved during the migration to AIX V4.1.4.

As a reminder, ACLs consist of:

- Base permissions:

  Base permissions are shown by the `aclget` command in the following format:

  ```
  attributes: SUID, SGID or STVX
  base permissions
      owner(username):   rwx
      group(groupname):  r-x
      others:            r--
  ```

  where:

  - SUID means setuid
  - SGID means setgid
  - SVTX means savetext (sticky bit)

  Base permissions are traditional UNIX permissions for the owner of the file, member of the group and others.

- Extended permissions:

  Extended permissions allow the owner of a file to define the access to the file more precisely. They extend the base file permissions (owner, group, other) by permitting, denying or specifying access modes for specific users or groups or for specific combinations of users and groups.

  The permit, deny and specify keywords are defined as follows:

  - permit: grants the user or group the specified access

  - deny: restricts access to the user or group

  - specify: precisely defines the file access

The following example shows the ACLs associated with the file called testfile. ACLs related to one file can be retrieved with the `aclget` command.

```
# aclget testfile
attributes:
base permissions
    owner(ewa):  rwx
    group(migra):  r-x
    others:  r--
extended permissions
    enabled
    deny     r--     u:colin,g:migra
    permit   r-x     u:yves,g:system
    permit   r-x     g:migra1
    specify  r--     u:koh,g:migra,g:migra1
```

In our example, users `ewa` and `colin` belong to the same group `migra`. User `koh` belongs to group `migra` and `migra1`.

The file testfile belongs to `ewa` who has read, write and execute permissions to this file. All members of the group `migra` have read and execute access to the file. Others have read-only access.

The extended permissions show that:

- User `colin` does not have read access to the file if he belongs to the group `migra`, which is the case.
- User `yves` has read and execute access to the file if he belongs to the group `system`.
- All members of the group `migra1` have read and execute access to the file.
- User `koh` only has read access to the file if he belongs to both `migra` and `migra1`, which is the case. This is equivalent to denying execute access to user `koh` if he belongs to both groups. This overwrites for user `koh` the overall execute permission granted to all members of groups `migra` and `migra1`.

We migrated our system from AIX V3.2.5 to AIX V4.1.3 and AIX V4.1.4 with the ACLs above. We could retrieve the ACLs after the migration.

Files with specified ACL permissions that have been archived with traditional UNIX archive programs, such as tar or cpio, lose their extended permissions after they are reloaded. This applies to both AIX Version 3.2 and AIX Version 4.1. The only archive programs that preserve the settings of the ACL extended permissions in both versions of AIX are backup and restore.

Attention!

Since AIX V3.2 system backup uses tar, your ACLs will not be saved by your system backup. You need to use the backup utility or aclget and aclput utilities to separately save the files on your system that have ACLs associated with them.

Note: For more information on ACLs and on AIX V3.2 and AIX V4.1 security issues, you should refer to these two redbooks:

- *Elements of AIX Security: R31*, GG24-3622-01
- *Elements of Security: AIX 4.1*, GG24-4433-00

1.2.8 Trusted Computing Base

The Trusted Computing Base (TCB) consists of a database and tools for checking the system integrity. The database is actually a file called sysck.cfg which stores various attributes of an object. A typical object is a file. Attributes of an object are one or more classes: the owner, the group, the mode, the type, the checksum value, and the size.

Following is an example of an entry in /etc/security/sysck.cfg:

```
/usr/sbin/exportvg:
        owner = root
        group = system
        mode = TCB,550
        type = FILE
        class = apply,inventory,bos.rte.lvm
        size = 7419
        checksum = "08716     8 "
```

The command `tcbck` allows you to check and update the database. The `tcbck -n ALL` command will check all the objects that are in the database.

Note: Never run `tcbck -y` to automatically fix anomalies found on an object in the database.

Attention!

The TCB installation is much more stringent in AIX Version 4.1 than in AIX Version 3.2. The TCB cannot be migrated from AIX V3.2 to AIX V4.1.

However, when migrating from AIX V3.2 to AIX V4.1.3 or AIX V4.1.4, it is **possible** to install the AIX V4.1 TCB during the migration. If you do so, the AIX V3.2 TCB will not be migrated, but you will be able to use the AIX V4.1 TCB after the migration. In section 1.4, "Migrating AIX" on page 33, you will find how to install the AIX V4.1 TCB.

Note: In AIX V4.1.2, you were not allowed to install the TCB when migrating.

1.2.9 Device Definition

When migrating to AIX V4.1, you should be aware of the following:

Attention!

External devices that are in an available or defined state prior to the migration and **not connected** or **powered off** will be removed from the ODM database during the migration process and the associated device drivers may not be installed.

This is true for detectable devices, such as tape drives, CD-ROMs, disk units, and so on. For example, if you configured a CD-ROM drive on your system and if this CD-ROM drive has been physically disconnected or powered off prior to the migration, its definition will be removed from the system.

If, before the migration, the CD-ROM is `Defined`:

```
# lsdev -C | grep cd
cd0     Defined  00-01-00-3,0 Multimedia CD-ROM Drive
```

Or, if the CD-ROM is `Available` before the migration, and powered off or disconnected during the migration, then after the migration, the CD-ROM is no longer there:

```
# lsdev -C | grep cd
#
```

We then recommend that you check on your system that all external devices you want to keep through the migration are in the `Available` state, connected AND powered on.

In order to document your device definitions, enter the following command:

```
# lsdev -C
```

1.2.10 Non-Root Volume Groups

If your system has user-defined volume groups, you should read the following sections.

1.2.10.1 Migration to AIX V4.1.3

If your system has user-defined volume groups, problems can occur after the migration from AIX Version 3.2.5 to AIX Version 4.1.3.

Attention 1!

After migrating to AIX V4.1.3, a non-root volume group which had its `AUTO ON` attribute set to `yes` prior to the migration will not be activated automatically even though its `AUTO ON` attribute is still set to `yes`. This problem is fixed in AIX V4.1.4

Attention 2!

After migrating to AIX V4.1.3, a non-root volume group which had its `AUTO ON` attribute set to `no` prior to the migration will have its attribute changed to `yes` and will be activated automatically.

This second problem comes from the fact that the `AUTO ON` attribute value is stored in the ODM database and not in the Volume Group Descriptor Area (VGDA). Thus, this information is not preserved during the migration.

In order to list all defined volume groups on your system, enter:

```
# lsvg
rootvg
nonrootvg
```

To check all active volume groups on your system, enter:

```
# lsvg -o
nonrootvg
rootvg
```

To check your `nonrootvg` volume group attributes, enter the following:

```
# lsvg nonrootvg
VOLUME GROUP:   nonrootvg                VG IDENTIFIER:   0001866927c909f0
VG STATE:       active                   PP SIZE:         4 megabyte(s)
VG PERMISSION:  read/write               TOTAL PPs:       75 (300 megabytes)
MAX LVs:        256                      FREE PPs:        61 (244 megabytes)
LVs:            2                        USED PPs:        14 (56 megabytes)
OPEN LVs:       2                        QUORUM:          2
TOTAL PVs:      1                        VG DESCRIPTORS:  2
STALE PVs:      0                        STALE PPs        0
ACTIVE PVs:     1                        AUTO ON:         yes
```

You can see in this example that nonrootvg is normally activated automatically at boot time (see the AUTO ON attribute).

When migrating to AIX V4.1.3, this non-root volume group will not be activated automatically. You will have to activate it manually at each boot. If you really need to migrate to AIX V4.1.3 instead of AIX V4.1.4, you will find below a temporary solution to this problem. We suggest that you explicitly vary on the non-root volume groups in the /etc/rc file:

- Save /etc/rc file into /etc/rc.save.

- Using your favorite editor, edit the /etc/rc file.

- Insert the following command into the /etc/rc file after the cfgvg command:

  varyonvg nonrootvg

```
#!/bin/ksh
# @(#)06 1.15  src/bos/etc/rc/rc.sh, cfgetc, bos411, 9428A410j 3/21/93 17:05:17
#
# COMPONENT_NAME: (CFGETC) Multi-user mode system setup
#
# FUNCTIONS: rc
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989, 1990
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
##############################################################

/usr/bin/dspmsg rc.cat 1 'Starting Multi-user Initialization\n'
PATH=/usr/sbin:/usr/bin:/usr/ucb:/etc::
ODMDIR=/etc/objrepos
export PATH ODMDIR

# Varyon all Volume Groups marked as auto-varyon.
# ( rootvg already varied on)
dspmsg rc.cat 2 ' Performing auto-varyon of Volume Groups \n'
cfgvg 1>&- 2>&-
varyonvg nonrootvg
```

Note: This temporary solution must only be made once the system has been migrated to AIX Version 4.1.3.

Note: Actually, the best way to overcome this non-root volume group problem is to deactivate the user-defined volume groups prior to the AIX Version 4.1.3 migration.

You will find below a summary of what happens during the migration to non-root volume groups when migrating to AIX V4.1.3.

Table 2. Migration to AIX V4.1.3.

Before Migration	After Migration
nonrootvg: AUTO ON = yes	**nonrootvg: AUTO ON = yes**

Defined Volume Groups
`#lsvg`
`rootvg`
`nonrootvg`
Active Volume Groups
`#lsvg -o`
`rootvg`
`nonrootvg`
	Defined Volume Groups
	`#lsvg`
	`rootvg`
	`nonrootvg`
	Active Volume Groups
	`lsvg -o`
	`rootvg`

nonrootvg: AUTO ON = no	**nonrootvg: AUTO ON = yes**

Defined Volume Groups
`#lsvg`
`rootvg`
`nonrootvg`
Active Volume Groups
`#lsvg -o`
`rootvg`
	Defined Volume Groups
	`#lsvg`
	`rootvg`
	`nonrootvg`
	Active Volume Groups
	`#lsvg -o`
	`nonrootvg`
	`rootvg`

1.2.10.2 Migration to AIX V4.1.4

When migrating to AIX V4.1.4, the first problem does not come up. All non-root volume groups are activated after the migration.

But the second behavior comes up. If a nonrootvg volume group is set to `AUTO ON: no` prior to the migration, its attribute will be set to `AUTO ON: yes` after the migration, and the volume group will be activated. The `AUTO ON` attribute is not preserved.

Below is a table that summarizes the non-root volume groups behavior when migrating to AIX V4.1.4.

Table 3 (Page 1 of 2). Migration to AIX V4.1.4.

Before Migration	After Migration
nonrootvg: AUTO ON = yes	**nonrootvg: AUTO ON = yes**

Defined Volume Groups:
`#lsvg`
`rootvg`
`nonrootvg`
Active Volume Groups
`#lsvg -o`
`rootvg`
`nonrootvg`
	Defined Volume Groups
	`#lsvg`
	`rootvg`
	`nonrootvg`
	Active Volume Groups
	`lsvg -o`
	`rootvg`
	`nonrootvg`

nonrootvg: AUTO ON = no	**nonrootvg: AUTO ON = yes**

Table 3 (Page 2 of 2). Migration to AIX V4.1.4.	
Before Migration	**After Migration**

Defined Volume Groups
`#lsvg`
`rootvg`
`nonrootvg`
Active Volume Groups
`#lsvg -o`
`rootvg`
	Defined Volume Groups
	`#lsvg`
	`rootvg`
	`nonrootvg`
	Active Volume Groups
	`#lsvg -o`
	`nonrootvg`
	`rootvg`

1.2.11 Migration Resource Requirements

In the following sections, you will find the disk space and memory requirements for the migration of AIX V3.2.5.

1.2.11.1 Disk Space Requirements

The Migration Install requires sufficient disk space in /tmp to save the configuration files and to run the `bosboot` command to build a boot image. It typically requires 16 MB. The Migration Install will remove all files in /tmp.

The Migration Install will check how much space is needed to save all the configuration files in /tmp and then check how much space is available in /tmp and on disks in rootvg.

If /tmp is only 8 MB, you need at least two free Physical Partitions (PPs) in rootvg in order to run the migration.

Attention!

If your /tmp file system is only 8 MB and you do not have any free partition available on disks in rootvg, the migration will hang before displaying the Migration Confirmation menu. No message will be displayed on the console. If this happens, press Enter. The system should reboot.

Once the system has checked that it has enough space to save the configuration files in /tmp, you will get the Migration Confirmation menu. If you go on migrating, the system will remove /dev/hd7 if it was used as the dump device. This will free up space that might be needed by installp process.

Also, the /(root) file system size must be at least 8 MB. If /(root) on your system is only 4 MB, the Migration Install will need at least one more available partition on your system to increase the size of root. In fact, if the root file system is only 4 MB, the migration process might run out of inodes in /.

Attention!

The migration might not succeed if your root file system is only 4 MB.

In fact, before starting a migration, we recommend that you have in rootvg this MINIMUM number of free partitions:

- Three free PPs if you have a 4 MB root file system, an 8 MB /tmp file system and a default (8 MB) /dev/hd7 used as the dump device

- Two free PPs if you have a root file system bigger than 4 MB, an 8 MB /tmp file system and a default (8 MB) /dev/hd7 used as the dump device

- Five free PPs if you have a 4 MB root file system, an 8 MB /tmp and no /dev/hd7 to be removed

If you do not have available space in rootvg, the only way to migrate your system using the Migration Install is first to recover some disk space in your file systems by cleaning the PTFs installed on your system or by removing some unused LPPs. For more information on how to recover some disk space on your system, refer to 1.2.12, "Disk Space Recovery."

Once the space is recovered, you must create the .fs.size file with the `mkszfile` command, edit this file and slightly reduce the size of the file systems in the file. You must then back up your system by using the `mksysb` command and restore it again on your system.

The AIX V4.1.4 for Clients package requires a minimum of 195 MB of disk storage to boot, install and execute without graphics. An additional 25 MB is required for graphics client systems.

The AIX V4.1.4 for Servers package requires a minimum of 220 MB of disk storage to boot, install and execute without graphics. An additional 30 MB is required for graphics client systems.

Note: After the migration, the disk space used by the system is not much bigger than the previous disk space used by AIX V3.2.5. In our tests, the AIX V3.2.5 system, including AIXwindows, used around 183 MB (result of the `du` command). After the migration, the disk space used by the system was 189 MB. This is due to the fact that some filesets are removed by the Migration Install, the PTF directories are cleaned up and the irrelevant microcode, device drivers and configuration methods are also removed

1.2.11.2 Memory Requirements
The memory requirement is 16 MB is you are migrating from a tape and only 8 MB if you are migrating from a CD-ROM.

Once running AIX V4.1.4, you might need to add some memory due to AIX V4.1 or new versions of LPPs or applications running in AIX V4.1. AIX V4.1.4 requires a minimum of 16 MB of memory. The amount of additional memory you need on top of this depends on the additional LPPs installed on your AIX V4.1 system.

1.2.12 Disk Space Recovery
The two procedures described below should be used with care and only if you are running out of the space needed to be able to run a migration.

1.2.12.1 Removing PTFs
You can also create disk space by using the ptfdir_clean shell script. This script was first made available with the release of the AIX Version 3.2.4 Preventive Maintenance Package (PMP). It is located in /usr/sbin. It will remove files and directories not needed by stand-alone systems. This script will erase the following files or directories:

```
/usr/lpp/<lppname>/inst_root
/usr/lpp/<lppname>/inst_<ptfid>
/usr/lpp/<lppname>/inst_<ptfid>.save
/usr/lpp/<lppname>/deinstl
/lpp/<lppname>/inst_<ptfid>
/lpp/<lppname>/inst_<ptfid>.save
/lpp/<lppname>/deinstl
/tmp/insutmp*
```

In order to be removed, PTFs must be either Committed or Available.

Attention!

Do not run the ptfdir_clean on a system you plan to use as a /usr SPOT or as a remote /usr server.

Check first how much space you can recover with the ptfdir_clean utility by entering the following command:

```
# ptfdir_clean -v
```

You will then get an output similar to this one:

```
Processing validity, and permissions ... Done
Processing /usr/lpp/<lppname>/inst_root ... Done
Processing list of ptfs ... Done
Processing other directories, may take several minutes ... Done

Total space that can be recovred using -y is approximately  15440 KB
Total space that can be recovred after commit all is approximately  0   KB

List of ptfs for directories that can be recovered using -y
     is in  /tmp/.recovered.island
List of lpps for which inst_root can be recovered using -y
     is in  /tmp/.product_recovered.island
```

If it is worth it, you can enter the following command to actually recover disk space:

```
# ptfdir_clean -yv
```

We ran this command on our system and saved approximately 17 MB of disk space in /usr.

1.2.12.2 Removing LPPs
In order to free some disk space, you can check if you really need to keep all of the installed LPPs. Sometimes, when installing a system, more LPPs are installed than what is absolutely necessary.

Removing a committed LPP is not possible through SMIT in AIX V3.2.5. You will find in Appendix D, "rmlpp Shell Script" on page 307 a shell script called `rmlpp` that will allow you to remove a specific, committed LPP on your AIX V3.2.5 system.

Note: For IBM employees, this shell script can be found in the AIXTOOLS. The package is called RMLPP.

This shell script will require a string of characters as argument and will find all the LPPs having this string of characters in their name. You will then be prompted with the LPP name found. You can then decide to remove the LPP or keep it.

Check first with the `lslpp -l` command which LPPs you would like to remove; then enter the following command:

```
# rmlpp <string>
```

where `<string>` is the LPP name or a string of characters within the LPP name.

For example, you can enter:

```
# rmlpp ate
```

You will be asked if you want to delete the bosext2.ate.obj fileset, which is the Asynchronous Terminal Emulator LPP.

Attention!

The rmlpp utility is not officially supported. Removing LPPs with the rmlpp shell script must be done very carefully and always after a system backup.

1.2.13 Documenting Your System

The following should be documented prior to the migration. Example outputs are shown below.

- Volume Groups:

```
# lsvg
rootvg
nonrootvg
```

- Volume Groups' Content:

```
# lsvg -p rootvg nonrootvg
rootvg:
PV_NAME          PV STATE    TOTAL PPs   FREE PPs    FREE DISTRIBUTION
hdisk0           active      95          13          06..03..00..00..04
nonrootvg:
PV_NAME          PV STATE    TOTAL PPs   FREE PPs    FREE DISTRIBUTION
hdisk1           active      75          68          15..12..11..15..15
```

- Logical Volumes:

```
# lsvg -o | lsvg -i -l
nonrootvg:
LV NAME              TYPE      LPs  PPs  PVs  LV STATE      MOUNT POINT
loglv00             jfslog    1    1    1    open/syncd    N/A
lv00                jfs       1    1    1    open/syncd    /data
rootvg:
LV NAME              TYPE      LPs  PPs  PVs  LV STATE      MOUNT POINT
hd6                 paging    16   16   1    open/syncd    N/A
hd5                 boot      2    2    1    closed/syncd  /blv
hd7                 sysdump   2    2    1    open/syncd    /mnt
hd8                 jfslog    1    1    1    open/syncd    N/A
hd4                 jfs       1    1    1    open/syncd    /
hd2                 jfs       50   50   1    open/syncd    /usr
hd1                 jfs       1    1    1    open/syncd    /home
hd3                 jfs       2    2    1    open/syncd    /tmp
hd9var              jfs       1    1    1    open/syncd    /var
```

- File systems:

```
# lsfs
Name              Nodename   Mount Pt         VFS  Size    Options  Auto Accounting
/dev/hd4          --         /                jfs  8192    --       yes  no
/dev/hd1          --         /home            jfs  8192    --       yes  no
/dev/hd2          --         /usr             jfs  409600  --       yes  no
/dev/hd9var       --         /var             jfs  8192    --       yes  no
/dev/hd3          --         /tmp             jfs  16384   --       yes  no
/dev/hd7          --         /mnt             jfs  --      --       no   no
/dev/hd5          --         /blv             jfs  --      --       no   no
/usr/bin/blv.fs   --         /usr/bin/blv.fs  --   --      --       no   no
/dev/lv00         --         /data            jfs  8192    rw       yes  no
```

- File systems size:

```
# df
Filesystem     Total KB    free %used    iused %iused Mounted on
/dev/hd4           4096     476   88%      663    64% /
/dev/hd9var        4096    3372   17%       98     9% /var
/dev/hd2         204800   15540   92%    12917    25% /usr
/dev/hd3           8192    7816    4%       33     1% /tmp
/dev/hd1           4096    3932    4%       17     1% /home
/dev/lv00          4096    3936    3%       16     1% /data
```

- Paging space:

```
# lsps -a
Page Space  Physical Volume   Volume Group    Size   %Used  Active  Auto  Type
hd6         hdisk0            rootvg          64MB      15    yes    yes    lv
```

- Dump device:

```
# sysdumpdev -l
primary             /dev/hd7
secondary           /dev/sysdumpnull
```

- Defined and available devices:

```
# lsdev -C
sys0       Available 00-00        System Object
sysplanar0 Available 00-00        System Planar
ioplanar0  Available 00-00        I/O Planar
bus0       Available 00-00        Microchannel Bus
sio0       Available 00-00        Standard I/O Planar
fda0       Available 00-00-0D     Standard I/O Diskette Adapter
siokta0    Available 00-00-0K     Keyboard/Tablet Adapter
sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2
scsi0      Available 00-01        SCSI I/O Controller
gda0       Available 00-03        Color Graphics Display Adapter
hdisk0     Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1     Available 00-01-00-1,0 320 MB SCSI Disk Drive
cd0        Available 00-01-00-3,0 Multimedia CD-ROM Drive
rmt0       Available 00-01-00-6,0 2.3 GB 8mm Tape Drive
cpucard0   Available 00-00        CPU card
proc0      Available 00-00        Processor
mem0       Available 00-0B        16 MB Memory Card
mem1       Available 00-0C        32 MB Memory Card
fd0        Available 00-00-0D-00  Diskette Drive
kbd0       Available 00-00-0K-00  United States keyboard
lvdd       Available              LVM Device Driver
lft0       Available              Low Function Terminal Subsystem
rootvg     Defined                Volume group
hd6        Defined                Logical volume
hd5        Defined                Logical volume
sysunit0   Available 00-00        System Unit
hd8        Defined                Logical volume
hd4        Defined                Logical volume
hd2        Defined                Logical volume
hd1        Defined                Logical volume
hd3        Defined                Logical volume
hd9var     Defined                Logical volume
sioma0     Available 00-00-0M     Mouse Adapter
ppa0       Available 00-00-0P     Standard I/O Parallel Port Adapter
ampx0      Available 00-04        X.25 CoProcessor/2 Adapter
mouse0     Available 00-00-0M-00  3 button mouse
pty0       Available              Asynchronous Pseudo-Terminal
inet0      Available              Internet Network Extension
lo0        Available              Loopback Network Interface
rcm0       Available              Rendering Context Manager Subsystem
aio0       Defined                Asynchronous I/O
nonrootvg  Defined                Volume group
loglv00    Defined                Logical volume
lv00       Defined                Logical volume
```

- Installed products:

  ```
  # lslpp -l
  ```

- Printer queues:

  ```
  # lpstat
  ```

1.2.14 Checking Root Authentication Method

You should make sure that the root user has a primary authentication method of
SYSTEM. To check the value, enter:

```
# lsuser -a auth1 root
root auth1=SYSTEM
```

If the value is different from SYSTEM, you can change it by entering:

```
# chuser auth1=SYSTEM root
```

1.2.15 Estimating Migration Times

The time taken by the AIX Migration Install process depends on the following factors:

- The installation method chosen

- The installation device (CD-ROM, 8mm tape, 4mm tape)

- The processor type and performance (POWER, POWER2, PowerPC)

- The amount of memory installed in the system

- The performance of the disks installed on the system

- The amount of software installed on the system

We migrated different systems and measured the AIX migration time using the Migration Install method. The following LPPs were installed on AIX V3.2.5 system:

```
X11fnt.coreX.fnt 01.02.03.00 AIXwindows Core X11 Fonts
X11fnt.ibm850.pc.fnt 01.02.03.00 AIXwindows Latin 1 (IBM-850) Fonts
X11fnt.iso88591.aix.fnt 01.02.03.00 AIXwindows Latin 1 (ISO8859-1) Fonts
X11fnt.oldX.fnt 01.02.03.00 AIXwindows Miscellaneous X Fonts
X11mEn_US.msg 01.02.03.00 AIXWindows NL Message files
X11rte.ext.obj 01.02.03.00 AIXwindows Run Time Environment Extensions
X11rte.motif1.2.obj 01.02.03.00 AIXwindows Motif 1.2 Run Time Support
X11rte.obj 01.02.03.00 AIXwindows Run Time Environment
bos.obj 03.02.00.00  The Base Operating System
bosext1.csh.obj 03.02.00.00 C Shell
bosext1.ecs.obj 03.02.00.00 Remote Customer Services
bosext1.extcmds.obj 03.02.00.00 Extended Commands
bosext1.mh.obj 03.02.00.00 Mail Handler
bosext1.uucp.obj 03.02.00.00 Unix to Unix Copy Program
bosext1.vdidd.obj 03.02.00.00 CGI Device Drivers
bosext2.acct.obj 03.02.00.00 Accounting
bosext2.ate.obj 03.02.00.00 Asynchronous Terminal Emulator
bosext2.dlc8023.obj 03.02.00.00 IEEE Ethernet (802.3) Data Link Control
bosext2.dlcether.obj 03.02.00.00 Standard Ethernet Data Link Control
bosext2.dlcfddi.obj 03.02.00.00 FDDI Data Link Control
bosext2.dlcqllc.obj 03.02.00.00 X.25 QLLC Data Link Control
bosext2.dlcsdlc.obj 03.02.00.00 SDLC Data Link Control
bosext2.dlctoken.obj 03.02.00.00 Token-Ring Data Link Control
bosext2.dosutil.obj 03.02.00.00 DOS Utilities
bosext2.games.obj 03.02.00.00 Games
bosext2.x25app.obj 03.02.00.00 X.25 Applications, API, Sample Source
bosnet.ncs.obj 03.02.00.00 Network Computing System
bosnet.nfs.obj 03.02.00.00 Network File System/NIS/RPC Libs & Utils
bosnet.snmpd.obj 03.02.00.00 Simple Network Management Protocol Daemon (Agent)
bosnet.tcpip.obj 03.02.00.00 TCPIP Applications
bsl.en_US.aix.loc 03.02.00.00 AIX Locale - English ( United States )
bsl.en_US.pc.loc 03.02.00.00 PC Locale - English ( United States )
bsmEn_US.msg 03.02.00.00 Base System Messages - U.S. English
```

Table 4 on page 30 gives some examples:

Table 4. AIX Migration Times.				
Model	**Memory**	**Disks**	**Device**	**Time**
320H	48 MB	400 MB + 320 MB	8mm tape	2 hours 10 minutes
320H	48 MB	400 MB + 320 MB	CD-ROM	1 hour 45 minutes
250	64 MB	1.0 GB	8mm tape	1 hour 45 minutes
250	64 MB	1.0 GB	CD-ROM	1 hour

These migration times do not include the entire system backup, including the non-root volume groups and the raw devices, if any.

1.3 Backing Up Your AIX V3.2.5 System

Even though the migration has been carefully planned, you must guard against any problem during or after the migration. Backing up the entire system is an absolute necessity. We assume that the backup you are going to create is to be restored on the same system having the same configuration. If it is to be restored on a different system or a system having a different configuration, the operation is called cloning. For more information on cloning, refer the *A Holistic Approach to AIX V4.1 Migration, Volume 2*, SG24-4653.

1.3.1 mksysb Limitations in AIX V3.2.5

These are some system backup limitations in AIX V3.2.5 when restoring a mksysb on the same system having the same configuration.

- `mksysb` removes the /dev directory completely and recreates it from the ODM database. If you created files in the /dev directory without any configuration methods to recreate them and without proper entries in the ODM database, these files will not be restored on the target system. This will affect most OEM device drivers.

- Console attributes are not archived as part of mksysb and therefore are not restored. For example, if you have the console login disabled prior to the system backup, it will be enabled after restoration.

- Restoration of an mksysb could result in two or more devices of the same type having their names swapped, and all of the customized attributes that go along with the names will also get swapped.

  When restoring mksysb, the devices database is recreated. All of the detectable devices (like tape drives, CD-ROM, 128 port adapters, and so on) are defined and configured from scratch. Then the mksysb defines the non-detectable devices (like TTYs, printers, modems and so on) and also updates the user-modified attributes of detectable devices.

  This can be a problem if, for example, you had on your system a token-ring card installed in slot 5 and later added a token-ring card in slot 4.

  When defining and configuring the detectable devices, the cards are configured in order starting with slot 1. Therefore, the card in slot 4 will be configured before the card in slot 5, and all the devices on card 4 will be configured before any devices on card 5. Thus, the cards will be named sequentially.

If the detectable devices are not named sequentially in your system, these devices will not be reconfigured properly. To check your adapter card device order, you can enter the following command:

```
# lsdev -C -s mca -F "location name" | sort
```

This problem could occur for adapter cards, tape drives, Remote Asynchronous Nodes (RANs), and so on. In many cases, the simplest solution to this problem is to swap any external cables or devices such as RANs to the adapters with the same names as the one they were originally attached to. Then update any system configuration documentation to reflect the new cabling.

- mksysb defines the non-detectable devices matching the parent names and types. If the parent is renamed as explained before, these non-detectable devices could end up at wrong locations. For example:

  Source machine has two 16-port cards; card 1 is in slot 5 and card 2 is added later and is in slot 4. So, the cards are named as follows:

  ```
  # lsdev -C -s mca -F "location name" | sort
  00-04   sa3
  00-05   sa2
  ```

  Now when mksysb restores, cards are named by going through each slot, so they will be named as:

  ```
  # lsdev -C -s mca -F "location name" | sort
  00-04   sa2
  00-05   sa3
  ```

  Since mksysb goes by name, all the devices (TTYs and printers) defined on card1 will appear on card2 and vice versa!

- If there is not enough space in the /tmp when mksysb was created, it can fail while restoring. At the very end of restoration, mksysb calls the `bosboot` command, which can fail if there is not enough space in /tmp.

  If this happens, you will get the following message:

  ```
  "0301-152 bosboot: not enough file space to create:"
  ```

  and will be prompted to go to maintenance mode or continue. Go to the maintenance mode, make some room in the /tmp by increasing the file system or by removing some files and then run the `bosboot` command.

Recommendation

For the most reliable results when creating an mksysb, the following steps should be taken:

- Power-on all devices and adapters on the system.

- Check that cards and adapters are named in sequential order on the system.

1.3.2 Backup Procedure

Only mounted file systems within rootvg are archived with the `mksysb` command. Thus, before backing up your system, determine which user file systems you want to keep mounted during the backup. In order to back up your system, do the following:

- At the command prompt, execute the following command to back up rootvg:

```
# smitty mksysb
```

The Backup the System menu will be displayed on the screen:

```
                              Backup the System

 Type or select values in entry fields.
 Press Enter AFTER making all desired changes.

                                                   [Entry Fields]
 WARNING:  Execution of the backup command will
           result in the loss of all material
           previously stored on the selected
           output medium. This command backs
           up only rootvg volume group.

    FORCE increase of work space if needed           no          +
  * Backup DEVICE or FILE                            []
  (example: /dev/rfd0)




 F1=Help            F2=Refresh          F3=Cancel          F4=List
 F5=Reset           F6=Command          F7=Edit            F8=Image
 F9=Shell           F10=Exit            Enter=Do
```

Specify the name of the backup device (for example /dev/rmt0) in the * **Backup DEVICE or FILE** field.

- After the mksysb has completed, back up the user-defined volume groups that contain the user applications and the user data on a separate media with the following command:

```
# find <search_directory> -print | backup -ivf <archive_media>
```

where the `<search_directory>` would be the directory on which the user file system is mounted, for example /data, and the `<archive_media>` would be your backup device, for example /dev/rmt0.1.

You can use the tar, cpio and backup utilities to back up non-root volume groups. But if you are using Access Control Lists (ACLs), tar and cpio will not preserve the ACL extended file permissions. In this case, you should use the `backup` command.

- If raw devices (logical volumes) have been defined on the system, you can archive them with the `dd` command. At this step, you should normally have already documented your system and kept track of the size of all logical volumes on your system by using the `lsvg` command.

To save a raw device, enter the following command:

```
# dd if=/dev/<rawdevice_name> of=/dev/rmtx
```

where `rmtx` is your tape drive name.

> **Note:** If you are using a tape drive as an output device, you may have to change your tape drive block size to 512 by using the following command:

```
# chdev -l rmtx -a block_size=512
```

> Or if the device block size is set to 1024, you may have to specify a `bs` parameter in your `dd` command like the following.

```
# dd if=/dev/<rawdevice_name> of=/dev/rmtx bs=1024
```

Note: File systems mounted across the network via the Network File System (NFS) will not be archived with the `mksysb` command.

1.4 Migrating AIX

Following is a flowchart that outlines the steps required to install the AIX Base Operating System (BOS) from CD-ROM or tape. The flowchart also shows how to chose the installation method.

Note: You will not need to select the installation method when migrating from AIX V3.2.5 to AIX V4.1 since the Migration Install is the default method.
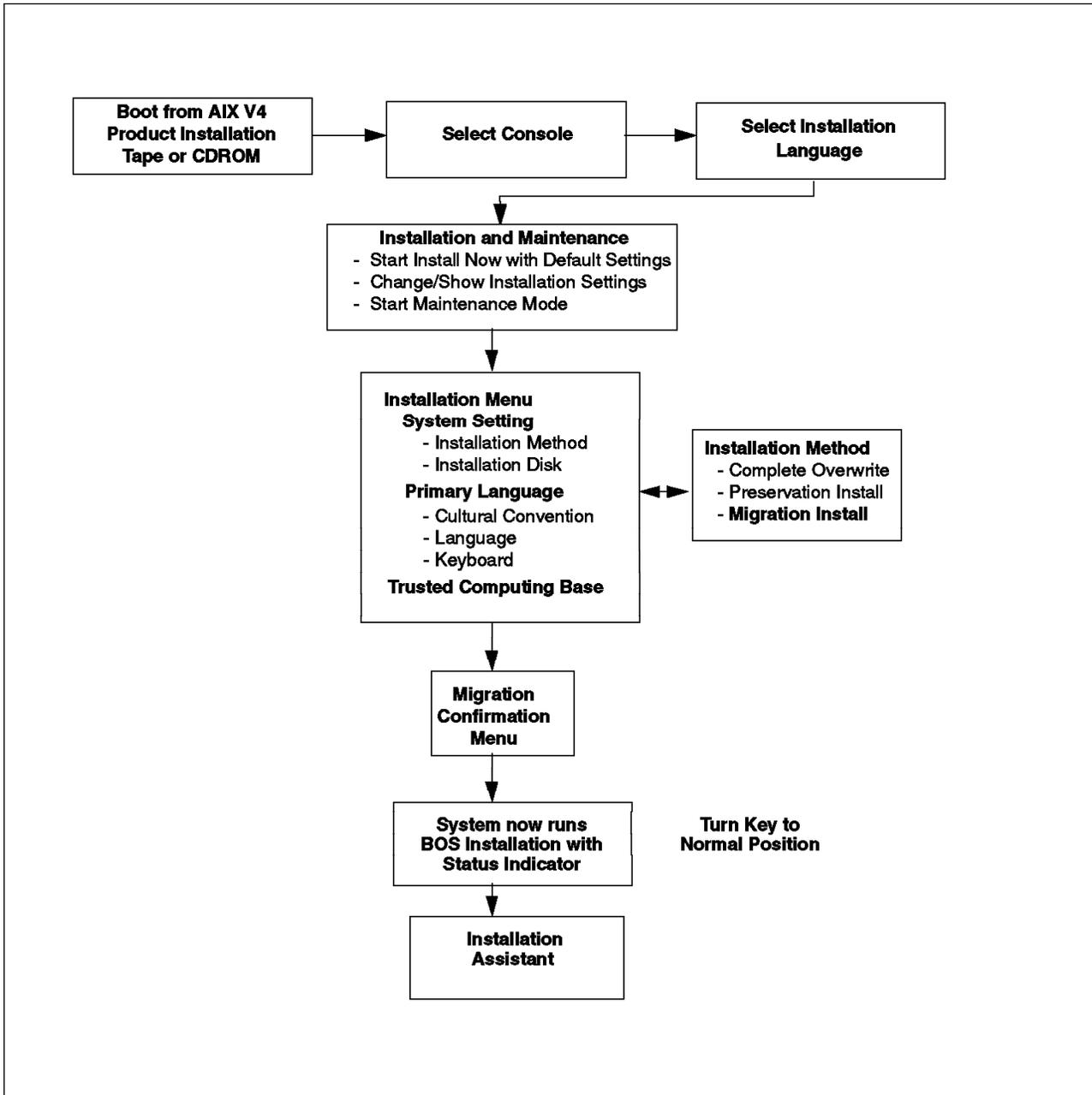
Figure 1. Migration Installation Flow

If your console is an ASCII terminal, before starting the Migration Install procedure, set the communication options of your terminal as follows:

```
Line Speed (baud rate) = 9600
Word Length (bits per character) = 8
Parity = no (none)
Number of Stop Bits = 1
Interface = RS-232C (or RS-422A)
Line Control = IPRTS
```

Note: The IPRTS flag allows an IBM 3151 terminal to ignore hardware handshaking.

Set the keyboard and display options on the ASCII terminal as follows:

```
Screen = normal
Row and Column = 24x80
Scroll = jump
Auto LF (line feed) = off
Line Wrap = on
Forcing Insert = line
Tab = field
Operating Mode = echo
Turnaround Character = CR
Enter = return
Return = new line
New Line = CR
Send = page
Insert Character = space
```

Following is the Migration Install procedure:

1. Insert the AIX Version 4.1 installation media into the installation device.

2. Turn the system key into the Service position.

3. Reboot the system with the following command:

   `# shutdown -Fr`

   By doing this, the system will boot from the installation media. The installation media can be a CD-ROM or a tape.

4. When the system has booted from the installation media, a screen is displayed that prompts you to select the installation language. Select the language in which the installation instructions and installation messages are to be displayed. The language selected for the installation does not have to be the same as the language you intend to use as a primary language environment.

5. The Welcome to the Base Operating System Installation and Maintenance menu is then displayed:

```
                    Welcome to Base Operating System
                       Installation and Maintenance



        Type the number of your choice and press Enter.


          1     Start installation Now with Default Settings
          2     Change/Show Installation Settings and Install
          3     Start Maintenance Mode for System Recovery






        88 Help ?



        Choice:
```

6. To confirm or change the installation and system settings that have been set for this installation, select option **2** and press **Enter**. The Installation and Settings screen will now be displayed:

```
                    Installation and Settings

 Either type 0 or press Enter to install with current settings, or type
 the number of the setting you want to change and press Enter.

      1.  System Settings :
              Method of Installation .............Migration
              Disk Where You Want to install......hdisk0

      2.  Primary Language Environment Settings (AFTER Install):
              Cultural Convention ................English (United States)
              Language ..........................English (United States)
              Keyboard ..........................English (United States)

      3.  Install Trusted Computing Base .........No

 >>> 0 Install with the settings listed above


    88 Help ?          +------------------------------------------------
    99 Previous Menu   | WARNING: Base Operating System Installation will
                       | destroy or impair recovery of SOME data on the
 >>> Choice [0]        | destination disk hdisk0
                       |
```

7. Since we are migrating from AIX V3.2.5 to AIX V4.1, the installation method is already set to Migration.

   Select the primary language environment settings. This language setting will be used for messages, keyboard map and cultural conventions.

8. Check that the disk(s) where you want to install is/are the disk(s) you intend to install on. The Migration Install will propose that you install the Base Operating System (BOS) on the disks belonging to rootvg. Since it is possible to have another rootvg on your system, these disks should belong to the rootvg you want to migrate. Check with the result of the `lsvg -p rootvg` command you used to document your system.

9. You can set the Install Trusted Computing Base option to Yes if you are migrating to AIX V4.1.3 or AIX V4.1.4. In this case, the AIX V3.2.5 TCB will not be preserved, but you will be able to use the AIX V4.1 TCB after the migration. If you set this option to No, you will not be able to use the AIX V4.1 TCB after the migration.

10. After all selections have been made, select option **0 Install with the settings listed above**. The Migration Menu Preparation in Progress screen will be displayed.

```
                    Migration Menu Preparation in Progress




     Please Wait.....








     Approximate            Elapsed Time
      %tasks                (in minutes)
     complete

        8                        1            Importing root volume group
```

11. After a few minutes, the Migration Confirmation Menu will be displayed on the
    console:

```
                    Migration Confirmation

 Either type 0 and press Enter to continue the installation, or type the
 number of your choice and press Enter.

        1  List the saved Base System configuration files which will not be
           merged into the system. These files are saved in /tmp/bos.

        2  List the filesets which will be removed and not replaced.

        3  List directories which will have all current contents removed.

        4  Reboot without migrating.

 >>>    0  Continue with migration.

       88 Help ?

 +----------------------------------------------------------------------
 WARNING: Selected files, directories, and filesets (installable options)
 from the Base System will be removed. Chose options 2 or 3 for more


   Select option [0]:
```

At this step, you can look at the base system configuration files which will not
be merged on the system and will be saved in /tmp/bos. You can also look at
the filesets which will be removed and not replaced by the migration and list
directories which will have all current contents removed.

The list of files mentioned in option 1 are stored in /tmp/bos/cfgfiles.moved.

The list of filesets mentioned in option 2 are stored in /tmp/bos/filesets.gone

The list of directories mentioned in option 3 are stored in
/tmp/bos/directories.gone

> **Note**
>
> It is recommended that you look at options 1, 2 and 3 before you continue with the migration. There could be some files, filesets or directories which have been overlooked when the migration was being planned.
>
> If for any reason you are not satisfied with the actions the system is going to take, you can decide to reboot the system without migrating and defer the migration to a later time.

After you have selected option **0 Continue with migration**, the following will appear on the display:

```
Saving system configuration files in /tmp/bos......
Removing obsolete filesets, directories, and files......
```

These two (2) lines will be displayed for a few minutes.

12. The Installing Base Operating System screen is then displayed:

```
                        Installing Base Operating System

 Turn the system key to the NORMAL position any time before the installation ends.




        Please Wait.....




     Approximate          Elapsed Time
        %tasks            (in minutes)
       complete


        25                     5           Restoring Base Operating System

```

13. Turn the system key to the NORMAL position.

At this point, the Output/Progress screen is displayed. This screen displays the following information:

- The elapsed time in minutes for each operation.

- The name of the operation currently running or the name of the product being installed.

- The percentage of tasks completed.

- Error messages, along with the product name being installed at the time the error occurred.

Note: During the installation, individual file names are no longer displayed.

14. When the migration has completed, the system will reboot if you changed the system key from the Service position to the Normal position. The Installation Assistant menu will then be displayed. If your system has a graphical display, you will see the following menu:
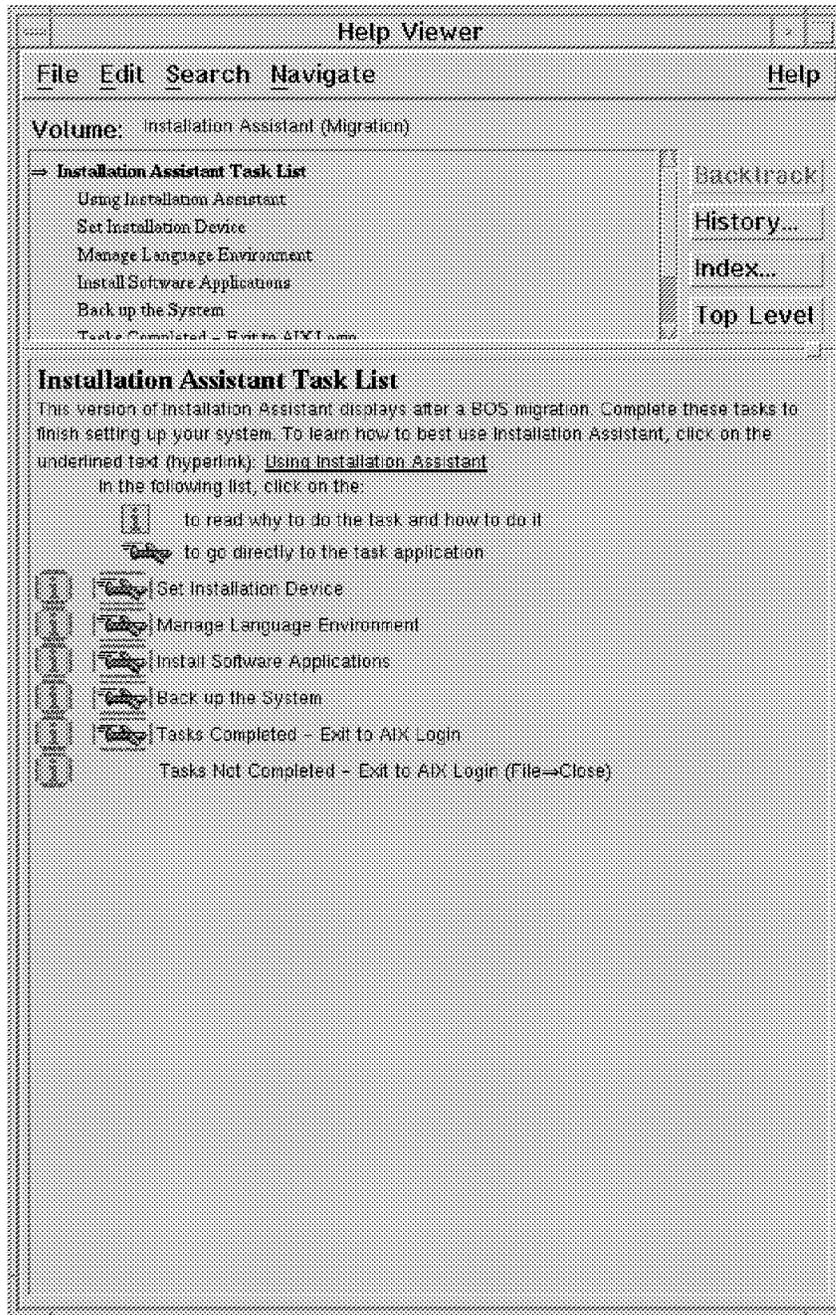
Figure 2. Installation Assistant Menu

1.5 Post Migration

After the migration, you will need to perform some tasks. The list of tasks required depend on your actual environment. Following are some examples:

- Check your system date and time.

- Check your system level and LPP consistency.

- Retrieve and check installation/migration messages.

- Check your configuration, and compare it with the documentation you saved.

- Check and set your paging space and dump device.
- Merge configuration files.
- Install new software products, to replace products that have been removed from AIX (for example, the C compiler).
- Install and customize iFOR/LS for AIX V4.1.4. Refer to the *AIX Version 4.1 iFOR/LS System Management Guide*, SC23-2665 for more information and also Chapter 2, "Installing iFOR/LS" on page 71 in this redbook.
- Install new versions of vendor application software that will run on AIX Version 4.1 including any OEM device drivers.
- Install your ported applications.
- Install your ported shell scripts.
- Restore scheduled jobs.
- Migrate your Motif/X11 environment to the Common Desktop Environment (CDE).

   Note: If you were using AIXwindows 2D (Motif and X11) in AIX Version 3.2.5, we recommend that you test your Motif/X11 environment in AIX Version 4.1.4 before completely moving your Motif/X11 environment to CDE. For more information on how to migrate from a Motif/X11 environment to CDE, see the redbook *A Holistic Approach to AIX V4.1 Migration, Volume 2*, SG24-4653.

- Test your entire system environment.
- Back up your AIX V4.1 system.

1.5.1 Checking the System Date and Time

Your system date might be changed during the migration. Check the date of the system by entering the following command:

```
# date
```

If the date and time is not correct, then change it by using the following SMIT fastpath:

```
# smitty date
```

1.5.2 Checking the System Level and LPPs

We recommend that you check your system level after the migration with the `oslevel` command. If the output does not show the AIX expected level (4.1.3.0 or 4.1.4.0), some prerequisite filesets may have been missing during the installation of the AIX V4.1 filesets. If you think you are supposed to see for example 4.1.4.0 and you do not, to see what is missing, you can run the following command:

```
# oslevel -l 4.1.4.0
```

Also, you can check the contents of the /var/adm/ras/devinst.log file. You will find more details on the installation messages in the next section.

Also run the `lppchk -v` command in order to see if all updates needed have been installed on your system.

1.5.3 Retrieving Installation Messages

This section explains how to retrieve some installation messages after the migration.

1.5.3.1 devinst.log File

The installp output goes to the file devinst.log, which is located in the /var/adm/ras directory. This file contains information on the software installed during the migration. It might be useful to look at it to see if all required software has been installed.

Following is an example of the devinst.log file:

```
BEGIN:Mon Dec  1 01:14:19 2025:120101141925
Command line is:
/usr/sbin/installp -V0 -b -acqNXQg -e /var/adm/ras/devinst.log -d/dev/rmt0.1 \
-f/../tmp/device.pkgs
+-----------------------------------------------------------------------------+
                Pre-installation Verification...
+-----------------------------------------------------------------------------+
installp:  Pre-installation verification may take several minutes.
done
WARNINGS
--------
  Problems described in this section are not likely to be the source of any
  immediate or serious failures, but further actions may be necessary or
  desired.

  Already Installed
  -----------------
  The number of selected filesets that are either already installed
  or effectively installed through superseding filesets is 40.  See
  the summaries at the end of this installation for details.

  NOTE:  Base level filesets may be reinstalled using the "Force"
  option (-F flag), or they may be removed, using the deinstall or
  "Remove Software Products" facility (-u flag), and then reinstalled.

  << End of Warning Section >>
:
```

1.5.3.2 BosMenus.log File

This file located in /var/adm/ras directory contains the migration menus and selections you have made during the migration. You can look and edit this file. Following is an example of the BosMenus.log file:

```
                        Installation and Settings

Either type 0 and press Enter to install with current settings, or type the
 number of the setting you want to change and press Enter.

    1  System Settings:
          Method of Installation.............Migration
          Disk Where You Want to Install.....hdisk0

    2  Primary Language Environment Settings (AFTER Install):
          Cultural Convention................C (POSIX)
          Language ..........................C (POSIX)
          Keyboard ..........................C (POSIX)


    3  Install Trusted Computing Base...... No

 >>> 0  Install AIX with the current settings listed above.

                        +-------------------------------------------------------
    88  Help ?          |    WARNING: Base Operating System Installation will
    99  Previous Menu   |    destroy or impair recovery of SOME data on the
                        |    destination disk hdisk0.
 >>> Choice [0]:    3
 :
```

1.5.3.3 installp Summary Report

Installation messages and information on products can be retrieved and checked
after the migration. Since the installp summary report has been extended in AIX
V4.1, more details can be obtained about what happened during the migration.

First of all, the installp Enhanced Summary Report records information in the ASCII
file /var/adm/sw/installp.summary. This file contains information about products that
have been requested to be installed. In AIX Version 3.2, this file included only the
list of products for which the installation process really began. Products that were
not installed because of problems found in the preinstallation part of installp did not
appear in the AIX V3.2 summary reports.

This file can be viewed with your favorite ASCII editor. You will find in this files
entries like this:

```
402:bos.rte:1::4.1.4.0:
```

Each entry has the following format:

<completion code>:<option name>:<final state>:<part>:<level>:

This format is documented in AIX V4.1 in the /usr/include/inuerr.h file. Following is
the description of each field:

- <completion code> is the result of the installation for the product.

- <option name> is the name of the product which has been installed or rejected.

- <final state> is the final state of the product.

- <part> is the part of product which has been installed or rejected.

- `<level>` is the level of the installed or rejected product

If the completion code is 0, this means that the installation has been successful. If it is 500, it means that the installation failed. A 4XX code indicates a pre-installation failure or a warning like 402 which says that the product is already installed. You will find all the pre-installation failure/warning codes in the /usr/include/inuerr.h file. This file is in the bos.adt.include fileset.

The following final states indicates that the product is:

- 1: available (AVAILABLE)
- 2: being applied (APPLYING)
- 3: applied (APPLIED)
- 4: being committed (COMMITTING)
- 5: committed (COMMITTED)
- 6: being rejected (REJECTING)
- 7: broken (BROKEN)
- 8: being deinstalled (DEINSTALLING)

The part is:

- U if usr part of the product
- R if root part of the product
- S if share part of the product

For example:

```
0:bosnet.tcp.client:5:U:4.1.4.0:
```

says that the usr part of the bosnet.tcpip.client product at level 4.1.4.0 had a successful installation with a final state of COMMITTED.

1.5.3.4 alog Command

AIX V4.1 provides a new command called alog. This function allows the output from installp to be recorded in the log file /var/adm/ras/bosinstlog.

The characteristics of the new alog command are the following:

- It is a circular logging program. It reads from sdtin and writes to stdout and copies the information into a file which has a fixed size. The logging program writes to the file, and when it is filled, new entries will be written over the oldest ones. The output does not include any time stamps, and the logging program does not provide any concurrency control (for example; the synchronization of multiple processes trying to log the same file at the same time).

- It is configurable. With alog, you can create or view log types, configure the log file size, verbosity and location. The default path for the alog files is /var/adm/ras.

alog can be viewed and configured using the following SMIT fastpath:

```
# smitty alog
```

The default alog log types can be seen with the following command:

```
# alog -L
boot
bosinst
nim
```

The default alog file size is 8192 bytes.

The following default log files are created by the installation process and stored in /var/adm/ras directory:

- bootlog: This file contains the boot messages. Its content can be retrieved with the following command:

  ```
  # alog -o -t boot
  ```

- bosinstlog: This file contains the messages from the BOS install process. Its content can be retrieved with the following command:

  ```
  # alog -o -t bosinst
  ```

  favorite editor.

- nimlog: This file exists if the migration was performed using NIM. It contains NIM installation messages.

Following is an example of the bosinstlog file:

```
# alog -o -t bosinst |pg
 Preparing target disks.
 Importing root volume group.
 rootvg
 Replaying the log.
 log redo processing for /dev/hd8
 syncpt record at 38bde8
 end of log 39f9b0
 syncpt record at 38bde8
 syncpt address 382564
 number of log records  = 1064
 number of do blocks = 67
 number of nodo blocks = 1
 Preserving old data.
 Copying old files .
 New volume on /dev/rmt0.1:
  Cluster 51200 bytes (100 blocks).
     Volume number 1
     Date of backup: Mon Oct  2 17:49:22 1995
     Files backed up by name
     User BUILD
     files restored: 1
 File System size changed to 32768
 Creating logical volumes.
 Mounting file systems.
 :
```

1.5.4 Paging Space and Dump Device

The Migration Install preserves the previous paging spaces. You might want to check that your previous paging spaces are still there by using the command:

```
# lsps -a
```

Also, you might want to increase the size of the paging space after the migration. You can do it from the Install Assistant menu at this end of the migration.

Note: Paging spaces outside the rootvg are not automatically activated after a migration and paging spaces in rootvg that were not activated prior to the migration are activated after the migration.

If your dump device was /dev/hd7, it is normally removed by the migration process. After the migration, the dump device is normally set to the paging space that is /dev/hd6.

You might want to have a dedicated dump device, especially if your system is used as a server. This will, of course, require some disk space to be available. First, check the estimated size of a dump image on your system with the following command:

```
# sysdumpdev -e
0453-041 Estimated dump size in bytes: 11294720
```

This command reports the size of your dump image in bytes. Then create a dedicated dump logical volume specifying a number of partitions that meets your need (3 in our case). Note that your partition size may be 4 MB or may be larger. Use the `lsvg rootvg` command and note the partition size of your rootvg.

```
# mklv -y hd7 -t sysdump rootvg 3
```

Then set your primary dump device to the new dedicated dump device:

```
# sysdumpdev -P -p /dev/hd7
```

Ensure that your system will automatically reboot after a crash:

```
# chdev -l sys0 -a autorestart=true
```

In this case, if a crash occurs, you will not be prompted by the system to load a tape drive in order to copy the dump on tape. The system will automatically reboot, and the dump will be stored on the dedicated dump device.

1.5.5 Obsolete Filesets

After the migration, if you issue the `lslpp -l` command, you will see some obsolete filesets. Obsolete filesets are AIX V3.2 filesets which have some but not all files replaced by filesets in AIX V4.1. These filesets cannot be updated, but are left on the system in case they are needed on the running system.

```
# lslpp -l | grep OBSOLETE
  X11fnt.coreX.fnt          1.2.3.0   OBSOLETE   AIXwindows Core X11 Fonts
  X11fnt.oldX.fnt           1.2.3.0   OBSOLETE   AIXwindows Miscellaneous X
  X11rte.ext.obj            1.2.3.0   OBSOLETE   AIXwindows Run Time
  X11rte.motif1.2.obj       1.2.3.0   OBSOLETE   AIXwindows Motif 1.2 Run Time
  X11rte.obj                1.2.3.0   OBSOLETE   AIXwindows Run Time
  bos.obj                   3.2.0.0   OBSOLETE   The Base Operating System
  bosext1.extcmds.obj       3.2.0.0   OBSOLETE   Extended Commands
  bsl.en_US.aix.loc         3.2.0.0   OBSOLETE   AIX Locale - English ( United
  bsl.en_US.pc.loc          3.2.0.0   OBSOLETE   PC Locale - English ( United
  bos.obj                   3.2.0.0   OBSOLETE   The Base Operating System
  bosext1.extcmds.data      3.2.0.0   OBSOLETE   Extended Commands
```

Attention!

You should not deinstall these filesets because they contain files that were not completely replaced by the AIX V4.1.4 equivalent filesets.

Removing one of these filesets can cause severe damage to your installation.

1.5.6 Merging System Configuration Files

Some configuration files must be merged manually after the migration. These files are actually saved in /tmp/bos during the migration.

You will find below a list of files from bos.rte that are not merged automatically by the migration process:

```
/etc/motd
/etc/rc
/etc/tsh_profile
/sbin/rc.boot
/usr/lib/objrepos/fix
/usr/lib/objrepos/fix.vc
/etc/security/sysck.cfg
/etc/consdef
/etc/dlpi.conf
/etc/inittab
/etc/magic
/etc/objrepos/CDiagDev
/etc/objrepos/Config_Rules
/etc/objrepos/DAVars
/etc/objrepos/FRUB
/etc/objrepos/FRUs
/etc/objrepos/MenuGoal
/etc/objrepos/TMInput
/etc/pse.conf
/etc/xtiso.conf
/usr/lib/objrepos/DSMenu
/usr/lib/objrepos/DSMOptions
/usr/lib/objrepos/DSMOptions.vc
/usr/lib/objrepos/GAI
/usr/lib/objrepos/GAI.vc
/usr/lib/objrepos/PdAt
/usr/lib/objrepos/PdAt.vc
/usr/lib/objrepos/PdCn
/usr/lib/objrepos/PdDv
/usr/lib/objrepos/PdDv.vc
/usr/lib/objrepos/XINPUT
/usr/lib/objrepos/XINPUT.vc
/var/adm/cron/at.deny
/var/adm/cron/log
/var/spool/cron/crontabs/adm
/var/spool/cron/crontabs/root
/var/spool/cron/crontabs/sys
```

The most two important files are /etc/inittab and /etc/rc.

Note: For each fileset outside of bos.rte, the migration of configuration files is handled separately. If there are configuration files which must be merged manually, then those files will be listed along with their stored locations at the end of the /var/adm/ras/devinst.log file. Search in this file for the pattern Post-installation Processing.

1.5.6.1 /etc/inittab File
You should be aware of the following:

Attention!

/etc/inittab will be reset to the default AIX V4.1 /etc/inittab during the migration.

The new standard AIX V4.1 inittab file looks like:

```
: @(#)49  1.28.2.7  src/bos/etc/inittab/inittab, cmdoper, bos411, 9430C411a 7/26/94
16:27:45
:
:   COMPONENT_NAME: CMDOPER
:
:   ORIGINS: 3, 27
:
:   (C) COPYRIGHT International Business Machines Corp. 1989, 1993
:   All Rights Reserved
:   Licensed Materials - Property of IBM
:
:   US Government Users Restricted Rights - Use, duplication or
:   disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
:
: Note - initdefault and sysinit should be the first and second entry.
:
init:2:initdefault:
brc::sysinit:/sbin/rc.boot 3 >/dev/console 2>&1 # Phase 3 of system boot
powerfail::powerfail:/etc/rc.powerfail 2>&1 | alog -tboot > /dev/console
rc:2:wait:/etc/rc 2>&1 | alog -tboot > /dev/console # Multi-User checks
fbcheck:2:wait:/usr/sbin/fbcheck 2>&1 | alog -tboot > /dev/console
srcmstr:2:respawn:/usr/sbin/srcmstr # System Resource Controller
rctcpip:2:wait:/etc/rc.tcpip > /dev/console 2>&1 # Start TCP/IP daemons
rcnfs:2:wait:/etc/rc.nfs > /dev/console 2>&1 # Start NFS Daemons
cron:2:respawn:/usr/sbin/cron
piobe:2:wait:/usr/lib/lpd/pio/etc/pioinit >/dev/null 2>&1  # pb cleanup
cons:0123456789:respawn:/usr/sbin/getty /dev/console
qdaemon:2:wait:/usr/bin/startsrc -sqdaemon
writesrv:2:wait:/usr/bin/startsrc -swritesrv
uprintfd:2:respawn:/usr/sbin/uprintfd
diagd:2:once:/usr/lpp/diagnostics/bin/diagd >/dev/console 2>&1
infod:2:once:startsrc -s infod
logsymp:2:once:/usr/lib/ras/logsymptom # for system dumps
tty0:2:respawn:/usr/sbin/getty /dev/tty0
tty1:2:off:/usr/sbin/getty /dev/tty1
```

The AIX V3.2.5 inittab file looks like:

```
: @(#)49  1.28  com/cfg/etc/inittab, bos, bos320 10/3/91
10:46:51
:   COMPONENT_NAME: CFGETC
:
:   ORIGINS: 3, 27
:
:   (C) COPYRIGHT International Business Machines Corp. 1989, 1990
:   All Rights Reserved
:   Licensed Materials - Property of IBM
:
:   US Government Users Restricted Rights - Use, duplication or
:   disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
:
: Note - initdefault and sysinit should be the first and second entry.
:
init:2:initdefault:
brc::sysinit:/sbin/rc.boot 3 >/dev/console 2>&1 # Phase 3 of system boot
powerfail::powerfail:/etc/rc.powerfail >/dev/console 2>&1 # d51225
rc:2:wait:/etc/rc > /dev/console 2>&1 # Multi-User checks
fbcheck:2:wait:/usr/lib/dwm/fbcheck >/dev/console 2>&1 # run /etc/firstboot
srcmstr:2:respawn:/etc/srcmstr        # System Resource Controller
rctcpip:2:wait:/etc/rc.tcpip > /dev/console 2>&1 # Start TCP/IP daemons
cons:0123456789:respawn:/etc/getty /dev/console
piobe:2:wait:/bin/rm -f /usr/lpd/pio/flags/* # Clean up printer flags files
cron:2:respawn:/etc/cron
qdaemon:2:wait:/bin/startsrc -sqdaemon
writesrv:2:wait:/bin/startsrc -swritesrv
uprintfd:2:respawn:/etc/uprintfd
rcncs:2:wait:sh /etc/rc.ncs
infod:2:once:startsrc -s infod
tty0:2:respawn:/etc/getty /dev/tty0
tty1:2:off:/etc/getty /dev/tty1
```

If you commented out, or removed, or added an entry into the inittab file, you might want to compare the saved file with the new ones.

For example, you can notice that the /etc/rc.ncs entry is no longer present in the inittab file. This entry will be added again when you will configure iFOR/LS.

Note: The following commands are the only supported methods for editing or modifying the inittab file:

- chitab: Changes records in the /etc/inittab file.

- lsitab: Lists records in the /etc/inittab file.

- mkitab: Adds records to the /etc/inittab file.

- rmitab: Removes records from the /etc/inittab file.

The main advantage in using these commands instead of a regular editor is that these commands will check the syntax of the entry being added or modified.

Following is an example on how to add an entry with mkitab. We supposed in this example that we need to start automatically an Oracle database at boot time.

```
# mkitab "oradb:2:once:su -oracle -c dbstart"
```

Once inittab has been modified, you can force init to re-read the file by entering:

```
# kill -1 1
```

or

```
# init q
```

This will send a sighup signal to init, and init will read the file again. If needed, the ODM will be updated at this time.

For more information on how to modify the inittab file, refer to the InfoExplorer documentation.

1.5.6.2 /etc/rc File
You should be aware of the following:

> **Attention!**
>
> /etc/rc will be reset to the default AIX V4.1 /etc/rc during the migration.

You will find below the default content of the AIX V4.1.4 /etc/rc configuration file.

```
#!/bin/ksh
# @(#)06  1.17  src/bos/etc/rc/rc.sh, cfgetc, bos41J, 9521A_all 5/22/95
18:09:30
#
# COMPONENT_NAME: (CFGETC) Multi-user mode system setup
#
# FUNCTIONS: rc
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989, 1995
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
###############################################################

/usr/bin/dspmsg rc.cat 1 'Starting Multi-user Initialization\n'
PATH=/usr/sbin:/usr/bin:/usr/ucb:/etc::
ODMDIR=/etc/objrepos
export PATH ODMDIR

# Varyon all Volume Groups marked as auto-varyon.
# ( rootvg already varied on)
dspmsg rc.cat 2 ' Performing auto-varyon of Volume Groups \n'
cfgvg 1>&- 2>&-

# Activate all paging spaces in automatic list
# (those listed in /etc/swapspaces)
dspmsg rc.cat 3 ' Activating all paging spaces \n'
swapon -a

# Perform file system checks
# The -f flag skips the check if the log has been replayed successfully
fsck -fp

# Perform all auto mounts
dspmsg rc.cat 4 ' Performing all automatic mounts \n'
mount all
```

```
# Remove /etc/nologin if left behind by shutdown
rm -f /etc/nologin

# Running expreserve to recover vi editor sessions
/usr/lib/expreserve - 2>/dev/null

# Write a dummy record to file /usr/adm/sa/sa<date> to specify
# that system start up has occurred.
#MSG= dspmsg rc.cat 7 'Write system start up record to /usr/adm/sa/sa'  date +%d
#echo $MSG
#/usr/bin/su - root -c /usr/lib/sa/sadc /usr/adm/sa/sa date +%d

# Manufacturing post install process.
# This must be at the end of this file, /etc/rc.
if [ -x /etc/mfg/rc.preload ]
then
     /etc/mfg/rc.preload
fi

dspmsg rc.cat 5 'Multi-user initialization completed\n'
exit 0
```

You might want to compare it with the previous AIX V3.2.5 version:

```
#!/bin/ksh
# @(#)06  1.13  com/cfg/etc/rc.sh, bos, bos320 4/30/91 14:25:11
#
# COMPONENT_NAME: (CFGETC) Multi-user mode system setup
#
# FUNCTIONS: rc
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1989, 1990
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
##################################################################

/usr/bin/dspmsg rc.cat 1 'Starting Multi-user Initialization\n'
PATH=/bin:/usr/bin:/usr/ucb:/etc::
ODMDIR=/etc/objrepos
export PATH ODMDIR

# Varyon all Volume Groups marked as auto-varyon.
# ( rootvg already varied on)
dspmsg rc.cat 2 ' Performing auto-varyon of Volume Groups \n'
/etc/cfgvg

# Activate all paging spaces in automatic list
# (those listed in /etc/swapspaces)
dspmsg rc.cat 3 ' Activating all paging spaces \n'
/etc/swapon -a
```

```
# Perform file system checks
# The -f flag skips the check if the log has been replayed successfully
fsck -fp

# Perform all auto mounts
dspmsg rc.cat 4 ' Performing all automatic mounts \n'
mount all

# Remove /etc/nologin if left behind by shutdown
rm -f /etc/nologin

# Running expreserve to recover vi editor sessions
/usr/lib/expreserve - 2>/dev/null

# Write a dummy record to file /usr/adm/sa/sa<date> to specify
# that system start up has occurred.
# dspmsg rc.cat 6 'Write system start up record to /usr/adm/sa/sa date '
#/bin/su - root -c /usr/lib/sa/sadc /usr/adm/sa/sa date +%d

# Manufacturing post install process.
# This must be at the end of this file, /etc/rc.
if [ -x /etc/mfg/rc.preload ]
then
     /etc/mfg/rc.preload
fi

dspmsg rc.cat 5 'Multi-user initialization completed\n'
exit 0
```

Check if you added entries in your AIX V3.2.5 /etc/rc file. These will have to be manually added to the AIX V4.1 /etc/rc file.

1.5.7 Testing Your New System

It is very difficult to test every single AIX Version 4.1 function, but there are some functions that are critical, and these functions should be tested.

Comparing the output of the migrated system with the output you got when documenting your system, you should make sure that:

- All user defined volume groups are varied on and can be accessed after the migration. Enter the following command to list the volume groups that are defined to the system:

```
# lsvg
rootvg
nonrootvg
```

List all actually active volume groups with the following command.

```
# lsvg -o
nonrootvg
rootvg
```

If one of the defined volume group is not active, check its AUTO ON attribute value.

- All previous logical volumes have been preserved using the command:

```
# lsvg -o | lsvg -i -l
nonrootvg:
LV NAME             TYPE       LPs   PPs   PVs   LV STATE       MOUNT POINT
loglv00             jfslog     1     1     1     open/syncd     N/A
lv00                jfs        1     1     1     open/syncd     /data
rootvg:
LV NAME             TYPE       LPs   PPs   PVs   LV STATE       MOUNT POINT
hd6                 paging     16    16    1     open/syncd     N/A
hd5                 boot       2     2     1     closed/syncd   /blv
hd8                 jfslog     1     1     1     open/syncd     N/A
hd4                 jfs        2     2     1     open/syncd     /
hd2                 jfs        55    55    1     open/syncd     /usr
hd1                 jfs        1     1     1     open/syncd     /home
hd3                 jfs        4     4     1     open/syncd     /tmp
hd9var              jfs        1     1     1     open/syncd     /var
```

- All file systems defined in /etc/filesystems are still there by using the following command:

  `# lsfs`

- All automounted file systems have been mounted and can be accessed after the migration. Enter the following command to list the file systems that have been mounted:

```
# mount
  node       mounted          mounted over    vfs      date          options
-------- ---------------  ---------------  ------ -----------  ---------------
         /dev/hd4         /                jfs    Dec 15 15:41 rw,log=/dev/hd8
         /dev/hd2         /usr             jfs    Dec 15 15:41 rw,log=/dev/hd8
         /dev/hd9var      /var             jfs    Dec 15 15:41 rw,log=/dev/hd8
         /dev/hd3         /tmp             jfs    Dec 15 15:41 rw,log=/dev/hd8
         /dev/hd1         /home            jfs    Dec 15 15:42 rw,log=/dev/hd8
         /dev/lv00        /data            jfs    Dec 15 15:42 rw,log=/dev/loglv00
```

- The size of the file systems. Note that in AIX V4.1, `df` returns the size in `512-blocks`. Use `df -k` to get the size in `1024-blocks` as in AIX V3.2.

```
# df
Filesystem    1024-blocks     Free %Used    Iused %Iused Mounted on
/dev/hd4             8192     3116   62%     1072    53% /
/dev/hd2           225280     5352   98%    15011    27% /usr
/dev/hd9var          4096     3200   22%      106    11% /var
/dev/hd3            16384    10928   34%       94     3% /tmp
/dev/hd1             4096     3824    7%       38     4% /home
/dev/lv00            4096     3936    4%       16     2% /data
```

- All required devices are configured and are available.

```
# lsdev -C
sys0        Available 00-00        System Object
sysplanar0  Available 00-00        System Planar
ioplanar0   Available 00-00        I/O Planar
bus0        Available 00-00        Microchannel Bus
sio0        Available 00-00        Standard I/O Planar
fda0        Available 00-00-0D     Standard I/O Diskette Adapter
siokta0     Available 00-00-0K     Keyboard/Tablet Adapter
sa0         Available 00-00-S1     Standard I/O Serial Port 1
sa1         Available 00-00-S2     Standard I/O Serial Port 2
scsi0       Available 00-01        SCSI I/O Controller
gda0        Available 00-03        Color Graphics Display Adapter
hdisk0      Available 00-01-00-0,0 400 MB SCSI Disk Drive
hdisk1      Available 00-01-00-1,0 320 MB SCSI Disk Drive
cd0         Available 00-01-00-3,0 Multimedia CD-ROM Drive
rmt0        Available 00-01-00-6,0 2.3 GB 8mm Tape Drive
cpucard0    Available 00-00        CPU card
proc0       Available 00-00        Processor
mem0        Available 00-0B        16 MB Memory Card
mem1        Available 00-0C        32 MB Memory Card
fd0         Available 00-00-0D-00  Diskette Drive
kbd0        Available 00-00-0K-00  United States keyboard
lvdd        Available              LVM Device Driver
lft0        Available              Low Function Terminal Subsystem
rootvg      Defined                Volume group
hd6         Defined                Logical volume
hd5         Defined                Logical volume
sysunit0    Available 00-00        System Unit
hd8         Defined                Logical volume
hd4         Defined                Logical volume
hd2         Defined                Logical volume
hd1         Defined                Logical volume
hd3         Defined                Logical volume
hd9var      Defined                Logical volume
sioma0      Available 00-00-0M     Mouse Adapter
ppa0        Available 00-00-0P     Standard I/O Parallel Port Adapter
ampx0       Available 00-04        X.25 CoProcessor/2 Adapter
mouse0      Available 00-00-0M-00  3 button mouse
pty0        Available              Asynchronous Pseudo-Terminal
inet0       Available              Internet Network Extension
lo0         Available              Loopback Network Interface
rcm0        Available              Rendering Context Manager Subsystem
aio0        Defined                Asynchronous I/O
nonrootvg   Defined                Volume group
loglv00     Defined                Logical volume
lv00        Defined                Logical volume
```

- All system users are defined and can log onto the system.

- The printer configuration still works, and you can print on all of the printers defined to the system.

- All applications and user data have been preserved.

- The network configuration works properly, and all network nodes can still be reached.

- All NFS file systems are mounted (will be shown by the `mount` command).

When the system environment has been verified, you can continue by testing your shell scripts and application(s).

Note: You will find more information on networking aspects in the redbook *A Holistic Approach to AIX V4.1 Migration, Volume 2*, SG24-4653.

1.5.8 Default Code Page

You should be aware that AIX Version 4.1 default code page is ISO8859-1 while AIX Version 3.2 default code page is IBM-850.

Users who want to configure their systems using a Latin 1 language environment and the IBM 850 code set can change their language environment after the BOS installation. This can be done by using SMIT and executing the following command:

```
# smitty chlang
```

The following screen will be displayed.

```
                    Change/Show Primary Language Environment

 Move cursor to desired item and press Enter.

   Change/Show Cultural Convention, Language, or Keyboard
   Select Set of Cultural Convention, Language, and Keyboard












 F1=Help              F2=Refresh          F3=Cancel           F8=Image
 F9=Shell             F10=Exit            Enter=Do
```

Select **Change/Show Cultural Convention, Language, or Keyboard**. Then you will get the following screen.

```
                Change/Show Cultural Convention, Language, or Keyboard

 Type or select values in entry fields.
 Press Enter AFTER making all desired changes.

                                                       [Entry Fields]
   Primary CULTURAL convention                       ISO8859-1  English (Un> +
   Primary LANGUAGE translation                      ISO8859-1  English (Un> +
   Primary KEYBOARD                                  ISO8859-1  English (Un> +
   INPUT device/directory for software               [/dev/rmt0]          +
   EXTEND file systems if space needed?               yes                 +




 F1=Help              F2=Refresh          F3=Cancel           F4=List
 F5=Reset             F6=Command          F7=Edit             F8=Image
 F9=Shell             F10=Exit            Enter=Do
```

Change your primary cultural convention, primary language translation and primary
keyboard to the IBM-850 code page. The product media must be installed in your
installation device so that IBM-850 filesets are automatically installed.

1.6 Backing Up your AIX V4.1 System

Once you have tested your AIX V4.1 system, we recommend at this step that you
back up your entire AIX V4.1 system.

The AIX Version 4.1 system backup utility is packaged in the bos.sysmgt.sysbr
fileset. This fileset should have normally been installed by default during the
migration. Check if it is installed by using the following command:

lslpp -L | grep bos.sysmgt.sysbr

The commands to make a bootable backup in AIX Version 4.1 are the same as
they were in AIX Version 3.2, but you should be aware of some differences,
enhancements and limitations of the system backup utility.

1.6.1 mksysb Enhancements

The mksysb utility will restore devices, such as the paging space(s), dump device
and mirrored data, without having to recreate them after restoration.

The AIX V4.1 mksysb utility will only save your rootvg volume group, but new
utilities are available to save and restore non-root volume groups. These utilities
are savevg and restvg.

A system backup is no longer created in a tar format but in a BFF (Backup File
Format) format. This means that ACLs are saved.

The mksysb utility allows you to select a list of files you do not want to be included
in your system backup. This is possible by editing, prior to the backup, the

/etc/exclude.rootvg file and inserting the names of the files that you do not want to be included into the system backup. Once the /etc/exclude.rootvg file has been created, you can create the system backup by specifying that you want to exclude files in the SMIT Backup the System menu:

```
# smitty mksysb
```

or by using the `mksysb` command with the `e` option as following:

```
# mksysb -e -i /dev/rmt0
```

Note: The `i` option creates the image.data file before starting the backup. If you want to manually change the image.data file, use the `mkszfile` command before using the `mksysb` command without the `i` option.

`mksysb` allows you to create MAP files when creating the backup. This will generate a layout mapping of the logical-to-physical partition for each logical volume in the volume group. This mapping is used to allocate the same logical-to-physical partition mapping when the image is restored.

A mksysb restoration can be customized by using ASCII files before creating the system backup. When the restoration of the system backup occurs, the system takes the information from these files and not from the standard input. This allows the system administrator to automate the restoration process. There are actually two files located under root (/):

- bosinst.data: This file contains installation parameters required at the target system, parameters that are normally set through the installation menus. It provides more flexibility when restoring the mksysb to the target system than previously with AIX V3.2. A sample of the bosinst.data file is provided in 3.8.1, "bosinst.data file" on page 160.

- image.data: This file is created by the `mkszfile` command that previously created the .fs.size file in AIX Version 3.2. This file contains information like the disk and logical volume layout, information on file systems and so on. This information is used by the BOS install process for creating rootvg. A sample of the image.data file is provided in the section 3.8.2, "image.data file" on page 163. Customizing this file will, for example, allow the system administrator to shrink the size of the file systems in order to save some space. In AIX V3.2, you had to edit the .fs.size file created by the `mkszfile` command and manually change the size of the logical volumes you wanted to reduce. The AIX V4.1 mksysb install process allows you to reduce the size of the logical volumes by setting the SHRINK stanza to yes in the image.data file, so you do not have to change stanzas.

> ┌─ **Attention!** ───
>
> Care must be taken when editing the image.data file. It is possible to make changes that will make the mksysb image uninstallable!

The default bosinst.data and image.data files can be found in the /var/adm/ras directory.

It is also possible to store the configuration files on a diskette. When installing from an mksysb image, the system checks if there is a diskette loaded that has any configuration files. If yes, the diskette configuration files will override the configuration files on tape. If no configuration files are found, the system will then search for them on a tape.

Note: If a diskette with configuration files is to be used, make sure that the diskette has a signature file. You can create a signature file by creating a file with vi and then insert the word data onto the first line. The utility that should be used for storing the configuration files on the diskette is the backup utility.

For example, copy all the configuration files you are using to / (root); change directory to root, and then execute the following command:

```
# ls ./bosinst.data ./signature ./image.data ./cust.file|backup -qiv
```

With a mksysb Install, the following will be preserved and restored:

- The blocksize of the file system (the fragmentation)
- The "nbpi" value (number of bytes per i-node)
- The file system compression policy
- The inter/intra policy for Logical Volumes

mksysb can be used in conjunction with Network Install Management (NIM). A NIM server can be used as a `mksysb` install media. In this case, you can define the configuration files for the mksysb Install as objects on the NIM server, and allocate them to the mksysb image.

mksysb no longer uses the startup fastpath (`smitty startup`). It can be started from the command line with the following command:

```
# mksysb /dev/rmt0
```

or by using SMIT with the following SMIT fastpath:

```
# smitty mksysb
```

1.6.2 mksysb Limitations in AIX V4.1

Limitations listed in 1.3.1, "mksysb Limitations in AIX V3.2.5" on page 30 also apply to AIX V4.1.

Recommendation

For the most reliable results when creating an mksysb, the following steps should be taken:

- Power-on all devices and adapters on the system.
- Cards and adapters must be named in sequential order on the system.

1.6.3 Backup Procedure

The procedure to back up your entire AIX V4.1 system is the following:

- Back up your rootvg volume group by using the following command:

  ```
  # smitty mksysb
  ```

- Back up your non-root volume groups by using the following command for each non-root volume groups.

  ```
  # savevg -i -f /dev/rmtx <nonrootvgname>
  ```

  where `/dev/rmtx` is your backup device name, and `<nonrootvgname>` is the name of your non-root volume group.

- Back up your raw devices with the following command:

  ```
  # dd if=/dev/<rawdevice_name> of=/dev/rmtx
  ```

  where `<rawdevice_name>` is the raw device name, and `/dev/rmtx` is your backup device name. You might need to specify a `bs` parameter to your `dd` command.

Note: Raw disk devices and raw logical volumes in rootvg are not saved by mksysb unless you specify that you want to create MAP files. When restoring the system, you will need to specify that you want to use these MAP files. In this case, raw disk devices and raw logical volumes created in rootvg will be preserved. Raw disk devices definitions and raw logical volumes definitions in non-root volume groups are saved by `savevg`, but the data on them is not - this must be done separately.

Attention!

This backup is intended to be restored on the same system having the same configuration. Due to some AIX V4.1.4 mksysb limitations, it might not be restored correctly on a system having a different configuration.

1.7 Exploiting AIX V4.1 Features

This section presents some AIX V4.1 features you might want to exploit after the migration.

Attention!

Do not use any AIX V4.1 specific features like disk striping, compressed file systems, greater than 2 GB file systems and so on in non-root volume groups unless you are sure that your migration is totally completed and you will not have to go back to AIX V3.2.

1.7.1 User and Group Definitions

If you would like your user definitions to take advantage of the new features in AIX Version 4.1, you will have to change the user definitions using the SMIT Change/Show Characteristics of a User menu; otherwise, all of the AIX V4.1 features will be set to system defaults when your system is migrated.

User definitions have been extended in AIX Version 4.1. The majority of the extensions are for Password Management and Login Controls.

Following is a list of the new Password Management options:

- Password triviality checking
- Settable password expiration warning
- Password reuse
- Password lifetime

Following is a list of the new Login Controls options:

- Restricted login time for a user
- Restricted login time for a port

- User invalidation after 'N' failed login attempts

- Port disabling after 'N' failed login attempts

- Login retry delay

- Login completion timeout

The password restrictions previously located in the file /etc/security/login.cfg have been moved to /etc/security/user. The password restrictions can now be set on a per-user basis instead of systemwide. In userconf.h, the pw_restrictions system attribute value has been removed in AIX Version 4.1. For more information, see the section called "The list of configuration files associated with the group security".

Following is a table comparing AIX Version 3.2 user definitions to AIX Version 4.1 user definitions:

Table 5 (Page 1 of 2). AIX V3.2 versus AIX V4.1 User Definitions.	
AIX V3.2.5	**AIX V4.1**
User NAME	User NAME
User ID	User ID
ADMINISTRATIVE User?	ADMINISTRATIVE USER?
PRIMARY Group	Primary Group
Group set	Group SET
ADMINISTRATIVE Groups	ADMINISTRATIVE GROUPS
Another user CAN SU to user?	Another user can SU TO USER?
SU Groups	SU Groups
HOME Directory	HOME directory
Initial PROGRAM	Initial PROGRAM
User information	User INFORMATION
EXPIRATION date (MMDDhhmmyy)	EXPIRATION date (MMDDhhmmyy)
	Is this user ACCOUNT LOCKED?
LOGIN User?	User can LOGIN?
User CAN RLOGIN?	User can LOGIN REMOTELY?
	Allowed LOGIN TIMES
	Number of FAILED LOGINS before user account is locked
	Login AUTHENTICATION GRAMMAR
Valid TTYs	Valid TTYs
	Days to WARN USER before password expires
	Password CHECK METHODS
	Password DICTIONARY FILES
	NUMBER of PASSWORDS before reuse
	WEEKS before password reuse
	Weeks between password EXPIRATION and LOCKOUT

Table 5 (Page 2 of 2). AIX V3.2 versus AIX V4.1 User Definitions.	
AIX V3.2.5	**AIX V4.1**
	Password MAX. AGE
	Password MIN. AGE
	Password MIN. LENGTH
	Password MIN. ALPHA characters
	Password MIN. OTHER characters
	Password MAX. REPEATED characters
	Password MIN. DIFFERENT characters
	Password REGISTRY
Max FILE Size	MAX. FILE size
Max CPU Time	MAX. CPU time
Max DATA Segment	MAX. DATA segment
Max STACK Size	MAX. STACK size
Max CORE File Size	MAX. CORE file size
File creation MASK	File creation UMASK
AUDIT classes	AUDIT classes
Trusted path	TRUSTED PATH?
PRIMARY authentication method	PRIMARY authentication method
SECONDARY authentication method	SECONDARY authentication method

The SMIT Security & Users menus have changed in AIX Version 4.1 compared to the SMIT menus in AIX Version 3.2.

At the command prompt, enter `smitty security`. This will display the Security & Users menu.

```
                          Security & Users

Move cursor to desired item and press Enter.

    Users
    Groups
    Passwords
    Login Controls













    F1=Help            F2=Refresh         F3=Cancel          F8=Image
    F9=Shell           F10=Exit           Enter=Do
```

The following menu is displayed if you select the Users option:

```
                                Users

Move cursor to desired item and press Enter.

    Add a User
    Change a User's Password
    Change / Show Characteristics of a User
    Lock / Unlock a User's Account
    Reset User's Failed Login Count
    Remove a User
    List All Users










    F1=Help            F2=Refresh         F3=Cancel          F8=Image
    F9=Shell           F10=Exit           Enter=Do
```
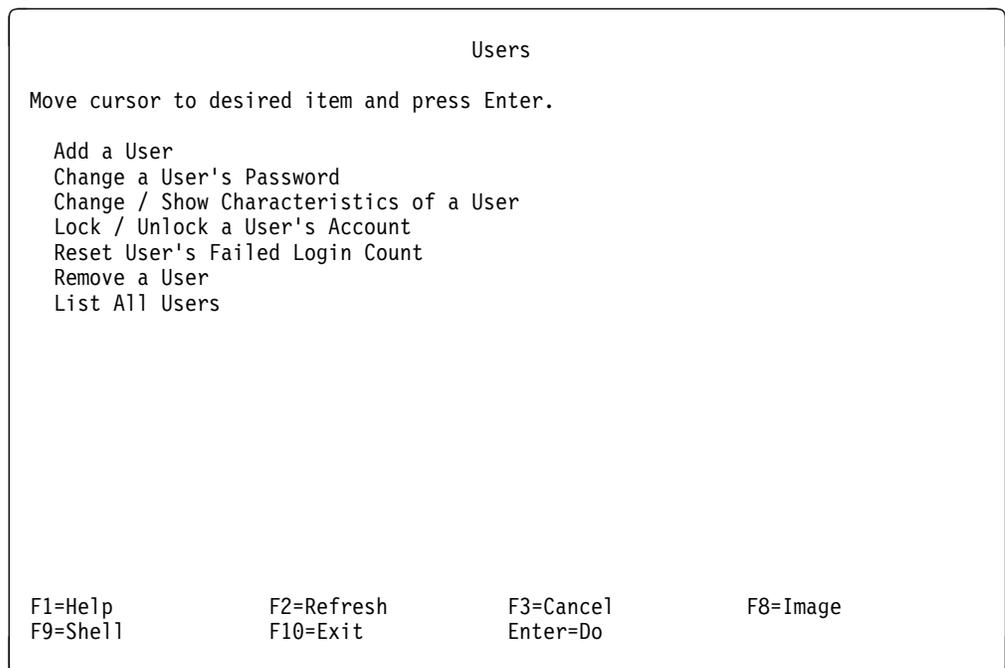
The following menu is displayed if you select the Groups option:

```
                                   Groups

Move cursor to desired item and press Enter.

   List All Groups
   Add a Group
   Change / Show Characteristics of a Group
   Remove a Group

















 F1=Help                 F2=Refresh              F3=Cancel               F8=Image
 F9=Shell                F10=Exit                Enter=Do
```
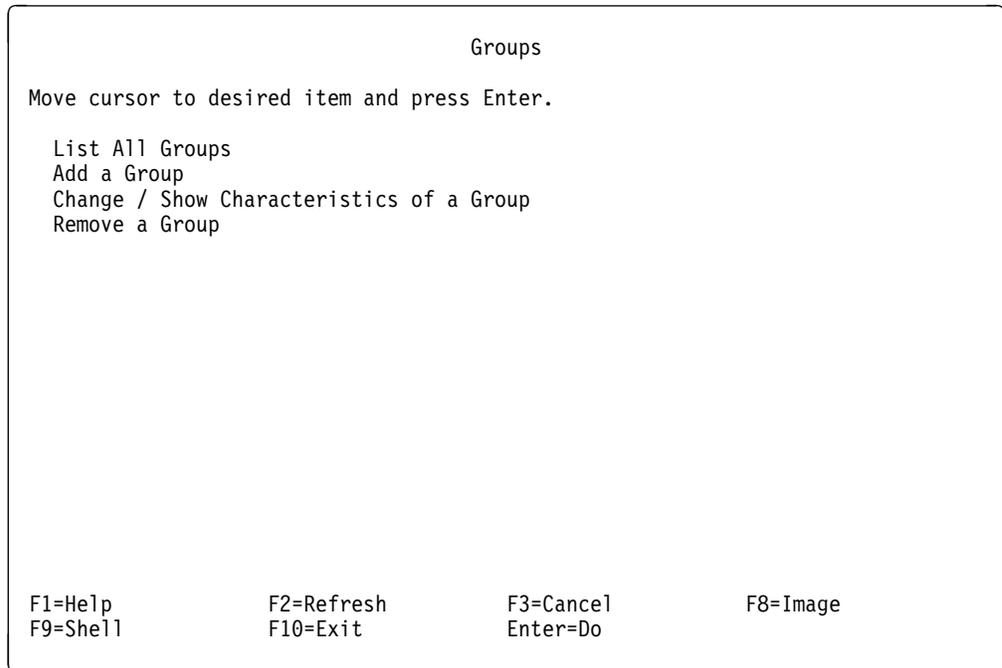
The following menu is displayed if you select the Passwords option:

```
                                  Passwords

Move cursor to desired item and press Enter.

   Change a User's Password
   Change / Show Password Attributes for a User

















 F1=Help                 F2=Refresh              F3=Cancel               F8=Image
 F9=Shell                F10=Exit                Enter=Do
```
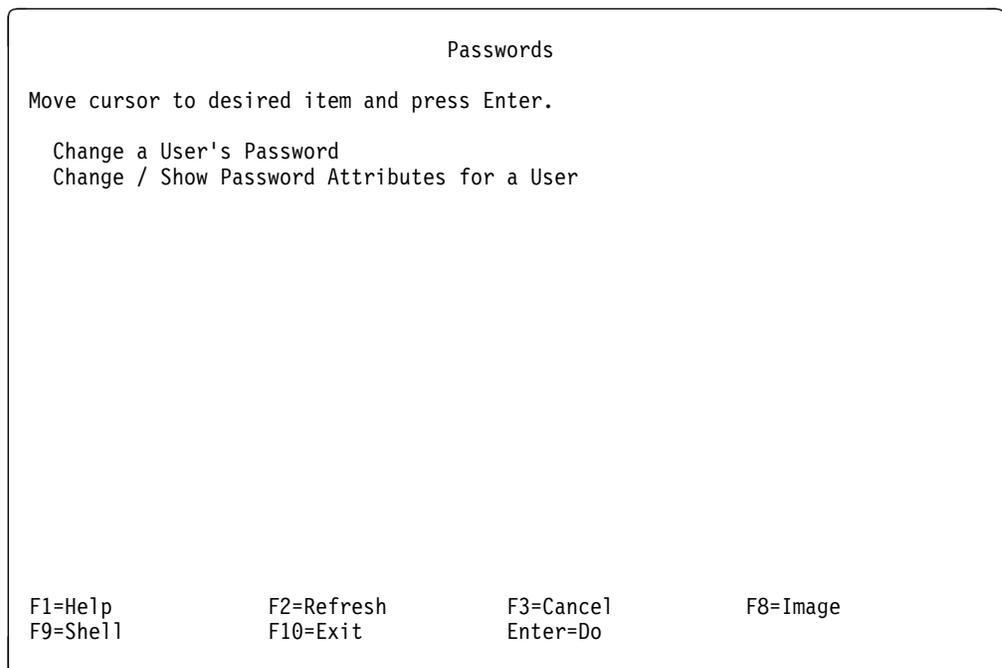
The following menu is displayed if you select the Login Controls option:

```
                              Login Controls

 Move cursor to desired item and press Enter.

   Change / Show Login Attributes for a User
   Change / Show Login Attributes for a Port












   F1=Help            F2=Refresh          F3=Cancel          F8=Image
   F9=Shell           F10=Exit            Enter=Do
```
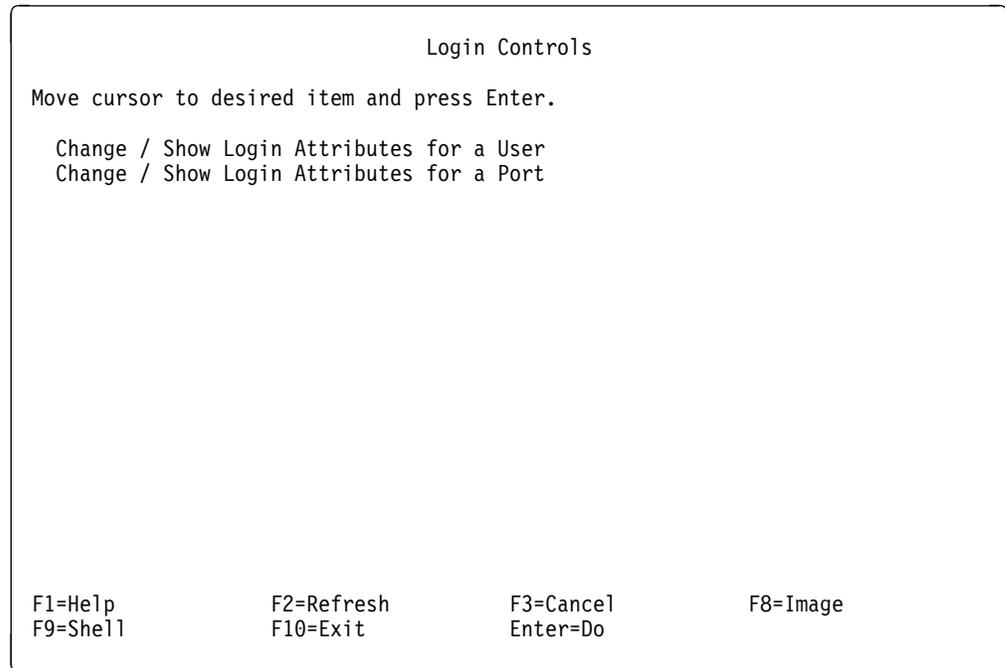
You can see from Table 5 on page 61, and from the SMIT menus, that there are some new security features that you might want to customize. For example, you might want to add some password check methods.

1.7.2 AIX V4.1 LVM and File Systems Features

These are some features related to the Logical Volume Manager and file systems in AIX V4.1 that you might want to use after the migration.

- Disk striping:

  When creating a logical volume, it is possible to specify a stripe size (4 KB, 16 KB, 32 KB, 64 KB, or 128 KB). If a stripe size is specified, the logical volume will be striped across the physical volumes specified at the logical volume creation. It means that blocks of data will be evenly distributed across those disks. This feature will increase sequential read and write throughput for applications issuing large I/O requests.

  There are some limitations in the current implementation:

  – A logical volume cannot be mirrored and striped at the same time.

  – The physical partitions for a striped logical volume must be evenly distributed among the disks.

  – After a logical volume has been created, striped size cannot be changed.

- Very large file systems (>2 GB):

  This feature allows system administrators to create a file system larger than 2 GB. The maximum limit for the current implementation is 64 GB per file system. However, the file itself still has the maximum limit of 2 GB.

- Compressed file systems:

  The data compression will be done by AIX and then written to hard disk. The physical disk usage may be less than the logical files size. Users will gain benefit from saving disk space, but it will cost processor time when reading or

writing a file to a compressed file system. Also, the compression factor is file data dependant.

- Fragmented blocks:

  This feature allows you to change the block size from the 4096-byte default to a smaller size that can be specified as 512, 1024 or 2048 bytes. File systems containing a large number of small files and directories will gain benefit from this space-saving feature.

1.7.3 File Systems Compatibility

The JFS (Journaled File System) in AIX Version 4.1 fully supports the file system images created under previous versions of AIX. When creating a file system under AIX V4.1, default options, such as the fragment size (4096), the number of bytes per inode (4096) and the compression algorithm (no), correspond to AIX V3.2 file system characteristics. Therefore, file systems created in AIX V4.1 with default options are compatible with AIX V3.2 BUT you must unmount the file systems on AIX V4.1 before going back to AIX V3.2 because the AIX V4.1 jfslog is not compatible with the AIX V3.2 jfslog.

File systems created with a different fragment size, a different number of bytes per inode or the compression option set to yes are incompatible with versions of AIX prior to AIX Version 4.1.

Backup and restore activities can be performed between file systems with different fragment sizes, but due to increased disk utilization, restore operations may fail because of a lack of free fragments.

Another reason for incompatibility between AIX Version 4.1 and AIX Version 3.2 file systems would be using a file system allocated on a striped logical volume.

1.8 Upgrading AIX V4.1 Client to AIX V4.1 Server

In general, if the Client package of AIX Version 4.1.X is installed on a system and there is a need to increase the number of licensed users, the Server package would have to be purchased.

Once you have received the Server package installation media, the Client system would have to be upgraded to enable you to increase the number of licensed users. To accomplish this task, the following fileset must be installed:

`bos.sysmgt.loginlic`

Once the fileset has been installed, you will be able to increase the number of licensed users via SMIT. The system will have to be rebooted after the number of licensed users has been increased. If the system is not rebooted, then the number of licensed users will remain at two.

Note: The bos.sysmgt.loginlic fileset will allow you to increase the mumber of designated users. If you bought licenses for networked users (or floating users) you will also need to install the iFOR/LS server fileset (bos.ifor_ls.server).

There are a few additional filesets that have been bundled with the Server package that are not available with the Client package.

1.8.1 Server vs. Client for AIX V4.1.3

Following is a table listing additional AIX V4.1.3 Server filesets compared to those found in AIX V4.1.3 Client.

Table 6. AIX V4.1.3 Server Additional Filesets.	
Fileset Name	**Fileset Description**
X11.x_st_mgr.compat 4.1.0.0	Xstation Manager Compatability
X11.x_st_mgr.rte 4.1.3.0	Xstation Manager Runtime Environment
bos.compat.NetInstl 4.1.0.0	AIX 3.2 Network Installation Tools
bos.info.en_US.aix_tech_ref 4.1.3.0	AIX Technical Reference Database - U.S. English
bos.info.en_US.adapters_gd_ref 4.1.	Adapters Database - U.S. English
bos.info.en_US.assemb_ref 4.1.3.0	Assembler Database - U.S. English
bos.net.nis.server 4.1.3.0	Network Information Service Server
bos.pci 4.1.1.0	AIX DOS Server for use with AIX Access for DOS Users
bos.sysmgt.loginlic 4.1.1.0	License Management
bos.sysmgt.quota 4.1.3.0	Filesystem Quota Commands
bos.sysmgt.nim.master 4.1.3.0	Network Install Manager - Master Tools
devices.mca.8f70.mpqp 4.1.3.0	4-Port Selectable Portmaster Adapter/A (MPQP) Software
devices.mca.8f70.rte 4.1.3.0	Portmaster Adapter/A Software
devices.mca.8f78.ucode 4.1.0.0	Serial Linked Disk Adapter/Controller/DASD Microcode
devices.mca.8f78.diag 4.1.3.0	Serial Linked Disk Adapter/Controller/DASD Diagnostics
devices.mca.8f78.rte 4.1.3.0	Serial Linked Disk Adapter/Controller/DASD Software
devices.mca.edd3.diag 4.1.0.0	16-Port Asynchronous Adapter EIA-422 Diagnostics
devices.mca.edd3.rte 4.1.3.0	16-Port Asynchronous Adapter EIA-422 Software
devices.mca.edd6.diag 4.1.0.0	16-Port Asynchronous Adapter EIA-232 Diagnostics
devices.mca.edd6.rte 4.1.3.0	16-Port Asynchronous Adapter EIA-232 Software
devices.mca.eff0.diag 4.1.0.0	X.25 CoProcessor/2 or Multiport/2 Adapter Diagnostics
devices.mca.eff0.rte 4.1.1.0	X.25 CoProcessor/2 or Multiport/2 Adapter Software
devices.scsi.scarray.rte 4.1.3.0	7135 RAIDiant Array DA Device Software Support
devices.scsi.scarray.diag 4.1.3.0	RAIDiant Array DA Device Diagnostics

1.8.2 Server vs. Client for AIX V4.1.4

Following is a table listing additional AIX V4.1.4 Server filesets compared to those found in AIX V4.1.4 Client.

Table 7. AIX V4.1.4 Server Additional Filesets.	
Fileset Name	**Fileset Description**
X11.x_st_mgr.compat 4.1.0.0	Xstation Manager Compatability
X11.x_st_mgr.rte 4.1.4.0	Xstation Manager Runtime Enviroment
bos.compat.NetInstl 4.1.0.0	AIX 3.2 Network Installation Tools
bos.info.en_US.adapters_gd_ref 4.1.4.0	Adapters Database - U.S. English
bos.info.en_US.aix_tech_ref 4.1.4.0	AIX Technical Reference Database - U.S. English
bos.info.en_US.assemb_ref 4.1.4.0	Assembler Database - U.S. English
bos.pci 4.1.1.0	AIX DOS Server for use with AIX Access for DOS Users
bos.sysmgt.loginlic 4.1.1.0	License Management
bos.sysmgt.nim.master 4.1.4.0	Network Install Manager - Master Tools
bos.sysmgt.quota 4.1.3.0	Filesystem Quota Commands
devices.isa.cxia128.diag 4.1.4.0	RISC PC ISA Async 128 Port Adapter Diagnostics
devices.isa.cxia128.rte 4.1.4.0	ISA 128-Port Asynchronous Adapter Software
devices.isa.cxia128.ucode 4.1.3.0	ISA 128-Port Asynchronous Adapter Microcode
devices.mca.8f70.mpqp 4.1.4.0	4-Port Selectable Portmaster Adapter/A (MPQP) Software
devices.mca.8f70.rte 4.1.4.0	Portmaster Adapter/A Software
devices.mca.8f78.diag 4.1.4.0	Serial Linked Disk Adapter/Controller/DASD Diagnostics
devices.mca.8f78.rte 4.1.4.0	Serial Linked Disk Adapter/Controller/DASD Software
devices.mca.8f78.ucode 4.1.0.0	Serial Linked Disk Adapter/Controller/DASD Microcode
devices.mca.edd3.diag 4.1.0.0	16-Port Asynchronous Adapter EIA-422 Diagnostics
devices.mca.edd3.rte 4.1.3.0	16-Port Asynchronous Adapter EIA-422 Software
devices.mca.edd6.diag 4.1.0.0	16-Port Asynchronous Adapter EIA-232 Diagnostics
devices.mca.edd6.rte 4.1.3.0	16-Port Asynchronous Adapter EIA-232 Software
devices.scsi.scarray.diag 4.1.4.0	RAIDiant Array DA Device Diagnostics
devices.scsi.scarray.rte 4.1.4.0	7135 RAIDiant Array DA Device Software Support

1.9 Recovery Procedure from a Failed Migration

If, for any reason, you decide to move back to AIX Version 3.2 because the migration failed, the application does not work properly or the OEM hardware device drivers are not fully compatible to AIX Version 4.1, you would have to reload AIX Version 3.2 from your mksysb tapes.

If you created any user-defined file systems with options different from the default one (blocksize different from 4096, and/or number of bytes per inode different from 4096, and/or compression enabled) or bigger than 2 GB, before you decide to return to AIX Version 3.2, you will have to back up those file systems, recreate them and restore them after you have restored your AIX Version 3.2.5 system backup.

If you changed an existing file system to be bigger than 2 GB, you will also need to back it up, and restore it after AIX V3.2.5 has been reinstalled.

If any changes have been made to the user data in any file systems, this data should be archived before any attempt is made to restore AIX Version 3.2.

All file systems in non-root volume groups using AIX V4.1 default options (options compatible with AIX V3.2 file systems) must be unmounted prior to restoring the AIX V3.2.5 system backup.

Restore your AIX V3.2 system backup by booting on your backup tape.

Attention!

When restoring a system backup, all devices should be powered on. This also includes any external adapters, such as the Remote Asynchronous Nodes (RANs) for the 128-port adapter.

After the first reboot following the AIX V3.2.5 mksysb installation, the shell script /usr/sbin/fbcheck will be invoked from /etc/inittab and executed.
This shell script invokes the /etc/firstboot routine which calls the rda command three times. The rda command is the command that reconstructs the proper mkdev and chdev commands to recreate the devices as they were on the machine where the mksysb was made. In /etc/firstboot, rda uses the saved device configuration database as input and generates files called reconfig1, reconfig2 and reconfig3 in /tmp. These files contain mkdev and chdev commands that create and alter devices. Then /etc/firstboot runs cfgmgr. After /etc/firstboot has been executed, it is renamed so that it will not be executed again.

If your terminal and printer definitions are missing after the restoration of the mksysb, you can use the commands within the reconfig files to restore the ODM definitions.

Make sure that the names used in the reconfig files match the devices and the names of the parent devices and that the locations match the current ODM.

Chapter 2. Installing iFOR/LS

After the migration to AIX V4, you may need to install new licensed products that require the iFOR/LS License System. The objective of this chapter is to help you in understanding iFOR/LS. You will also learn how to set up:

- Nodelocked licenses on a stand-alone machine

- Floating license on a license server and several client machines

For more complex configurations, we recommend that you refer to the iFOR/LS documentation.

2.1 Related Publications

This chapter is only a summary of the iFOR/LS implementation and configuration. For more detailed information, please consult the following documents:

- *AIX Version 4.1 iFOR/LS System Management Guide*, SC23-2665

- *AIX Version 4.1 iFOR/LS Tips and Techniques*, SC23-2666

The following documents are also available from Gradient Technologies Inc:

- *NetLS Quick Start Guide*

- *Network Licensing*

- *Managing Software Products with NetLS*

- *Managing NCS Software*

- *Licensing Software Products with NetLS*

See also "Related Publications" on page xxi for publications related to AIX.

2.2 iFOR/LS Overview

iFOR/LS is an acronym for Information for Operation and Retrieval/License System. It is a superset of the product NetLS. iFOR/LS, plus all upgrades and enhancements, is available through both IBM and Gradient Technologies, Inc.

iFOR/LS is an application for managing and distributing software licenses electronically and protecting the assets on the CD-ROM or tape installation media. The primary purpose of iFOR/LS is for software license compliance.

Administrators can manage all license servers on a network from a single workstation without remote logins, making iFOR/LS ideal for large networks.

Hewlett-Packard (HP) acquired NetLS from Apollo Computer, Inc. in 1987. Gradient Technologies, Inc. bought derivative rights from HP to use the NetLS code. Gradient has been upgrading the functionality of NetLS for some time and decided to use a new name, iFOR/LS, to indicate the enhancements it has added. The enhancements are pervasive and include changes to the server, the runtime code, NCS, the Graphical User Interfaces (GUIs), and the functionality of the GUIs.

Both NetLS and iFOR/LS are produced by Gradient, and both are associated with copyrights from Gradient, Apollo, and HP. Gradient is the sole owner of the new trademark, iFOR/LS. While the name and functionality of the product is iFOR/LS, the iFOR/LS name is not universal. For instance, the file system paths and binary names will be netls. You can still expect to find the nodelock file in /usr/lib/netls/conf/nodelock.

iFOR/LS can be run on other platforms besides AIX (for instance HP/UX8, HP/UX9, SunOS4.1, DECstation, NeXT); so the network may contain a mixture of system types.

Since each license server operates independently, administrators can partition licenses among multiple servers, depending on their needs. For example, if two products have 10 licenses each, an administrator may choose to install all 20 licenses on one server for easy administration. Alternatively, the administrator can configure two servers with five licenses for each product on each server to balance license requests and increase license server availability.

2.3 iFOR/LS License Types

There are several types of software licenses. Following are some of the most common license types.

- Nodelocked
- Concurrent-use
- Use-once
- Compound License
- License Combinations
- Developer Licences

Note: Concurrent-use, Use-once, Compound License, License Combinations, and Developer Licenses all require the iFOR/LS server code to operate.

2.3.1 Nodelocked

A nodelocked license restricts an application's use to a particular machine. Having both the application and the license installed on the same machine eliminates the overhead of a license server since the nodelocked license key is entered locally. Nodelocked licenses allow an unlimited number of uses but only on the machine where the license key is installed. When the software product starts up, it checks the file /usr/lib/netls/conf/nodelock on the current node to see if an appropriate entry exists that would allow access. This license type does not use the network, Network Computing System (NCS) or the license server.

2.3.2 Concurrent-use

Concurrent-use licenses are "floating" licenses that allow users to run software over a network. Typically, the application is installed on one machine on the network, and multiple instances of the application can be executed concurrently over the network. The customer purchases and installs the number of licenses expected to be used at peak periods.

When a software product is started, it communicates with the license server (`netlsd` daemon) on the network to determine if concurrent licenses for that product have been configured and whether any free licences are available.

iFOR/LS requires any software product using a concurrent-use license to periodically inform the `netlsd` daemon that it is still active. This prevents products that have halted abnormally from holding licenses they no longer need.

2.3.3 Use-once

A use-once license allows users to run an application one time. After the application has been used, the license server will not allow the license to be used again. One utilization is the "try-and-buy" scenario in which a customer obtains demonstration software with several use-once licenses. The administrator installs the licenses on a server, and the application can be invoked for a predetermined number of times.

Each time a user invokes the software product, the number of use-once licenses available is decremented. These licenses are useful as free-sample or demonstration licenses. They are also useful as backup licenses on products that may experience "peak usage" draws. For example, if the number of users exceeds the number of concurrent-use licenses available, a customer may want a few use-once licenses available for use during this peak time. This can be monitored, and if a pattern develops, more concurrent-use licenses may be purchased.

2.3.4 Compound License

Compound licenses bundle a group of iFOR/LS licenses for later redistribution. When a compound license is received, an administrator can generate a limited number of licenses for target machines. For example, a software distributor can purchase 50 nodelock licenses bundled into a single compound license. Or a large company can request a compound license containing 70 concurrent-use licenses for a particular application. Administrators can then redistribute these licenses to servers on the company's network as they see fit.

2.3.5 License Combinations

License combinations provide great flexibility. A common form combines a concurrent-use license and a nodelock license. This ties software use to a specific machine while limiting the number of concurrent uses.

2.3.6 Developer Licenses

Developer licenses enable the license generator portion of the ADK (iFOR/LS Application Developer's Toolkit) to create licenses for testing. License generation centers also use them to fulfill customer requests for end-user licenses.

2.4 iFOR/LS Concepts

Figure 3 on page 74 shows how the iFOR/LS licensing system works. When a user invokes a product, the product requests a license from the license server daemon (`netlsd`). The license server then checks in the license database if a valid license is available. It also checks in the user file if that user is allowed to use that product.

Note: The user file is optional.

If a license is available and the user is allowed to use the product, then the user can start running the product. When the user exits the product, the product releases the license back to the license server.
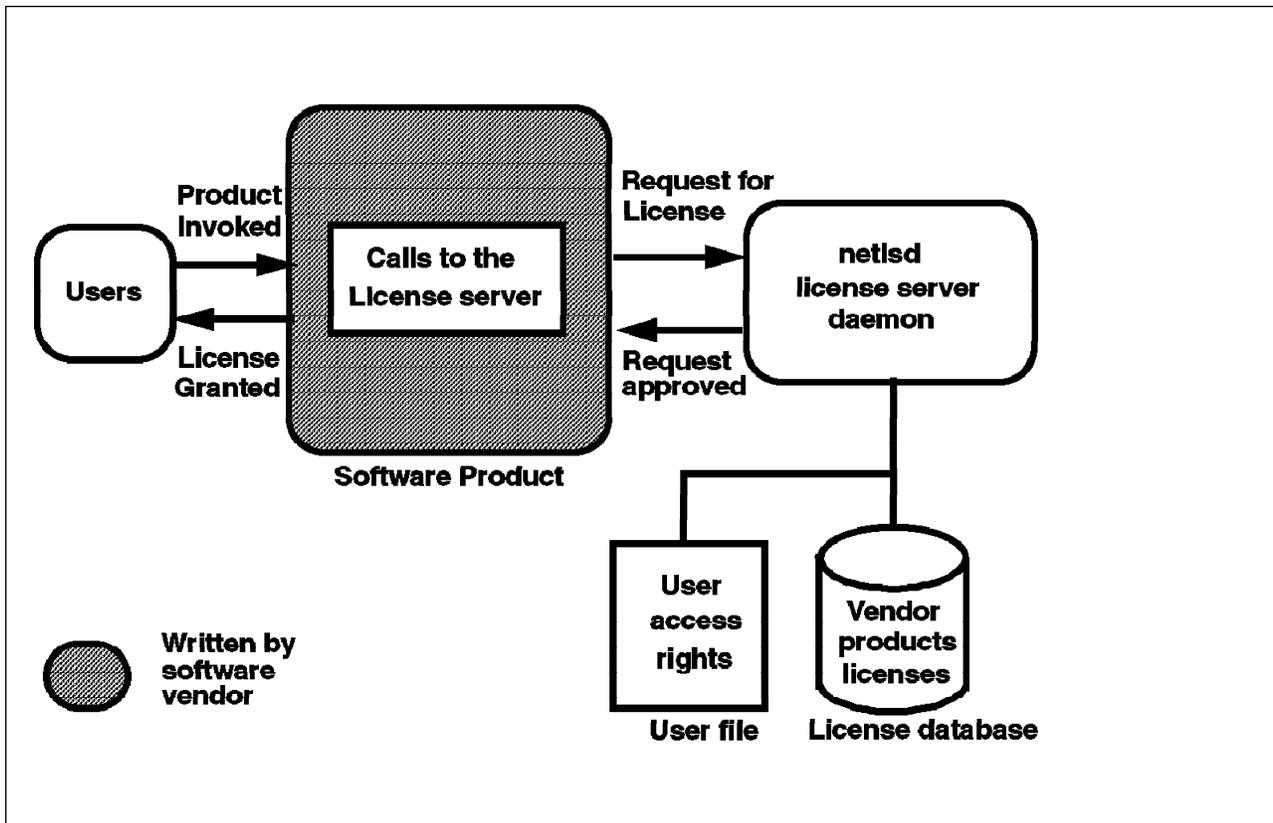
Figure 3. iFOR/LS License Server Operation

iFOR/LS is layered on top of NCS. This means that NCS provides a way for client machines to locate the license server on the network. In order to locate the license server, the client system uses a service called the Location Broker. The Location Broker is composed of the Local Location Broker and the Global Location Broker.

The Global Location Broker maintains a database containing all the services available on the network. The Local Location Broker is aware of which service is available on the machine on which it is running. The Local Location Broker asks the Global Location Broker when it does not know where to locate a service requested by a user.

The Local Location Broker daemon is `llbd`, and the Global Location Broker daemon is `glbd`.

When you deal with iFOR/LS and want to set up concurrent-use licenses, you need to understand what an NCS cell is and the difference between a default cell and an alternate cell.

- **NCS Cell:** An NCS cell is a group of systems on the network serviced by the same Global Location Broker (`glbd` daemon). There is no correspondence between cells and the network topology. Two hosts close to each other can belong to different NCS cells, and two remote hosts can belong to the same cell.

- **UUID:** Each cell is identified by a Universal Unique Identifier (UUID).

There are two different kind of cells:

- **Default Cell:** This cell uses a default identifier. A default cell allows access to a service from any node in the network that does not belong to an alternate cell. This is the most common cell used in iFOR/LS configurations. A system belonging to the default cell does not need the file glb_obj.txt.

- **Alternate Cell:** An alternate cell restricts access to a specific group of network nodes. Each node must have the same UUID stored in a file called glb_obj.txt.

Figure 4 shows a configuration with a default cell and an alternate cell. You can see that in the alternate cell, the daemons `glbd` and `llbd` run on the server. All the systems belonging to the same alternate cell must have the glb_obj.txt file containing the same UUID. This UUID is generated when NCS is configured. It can also be generated manually.
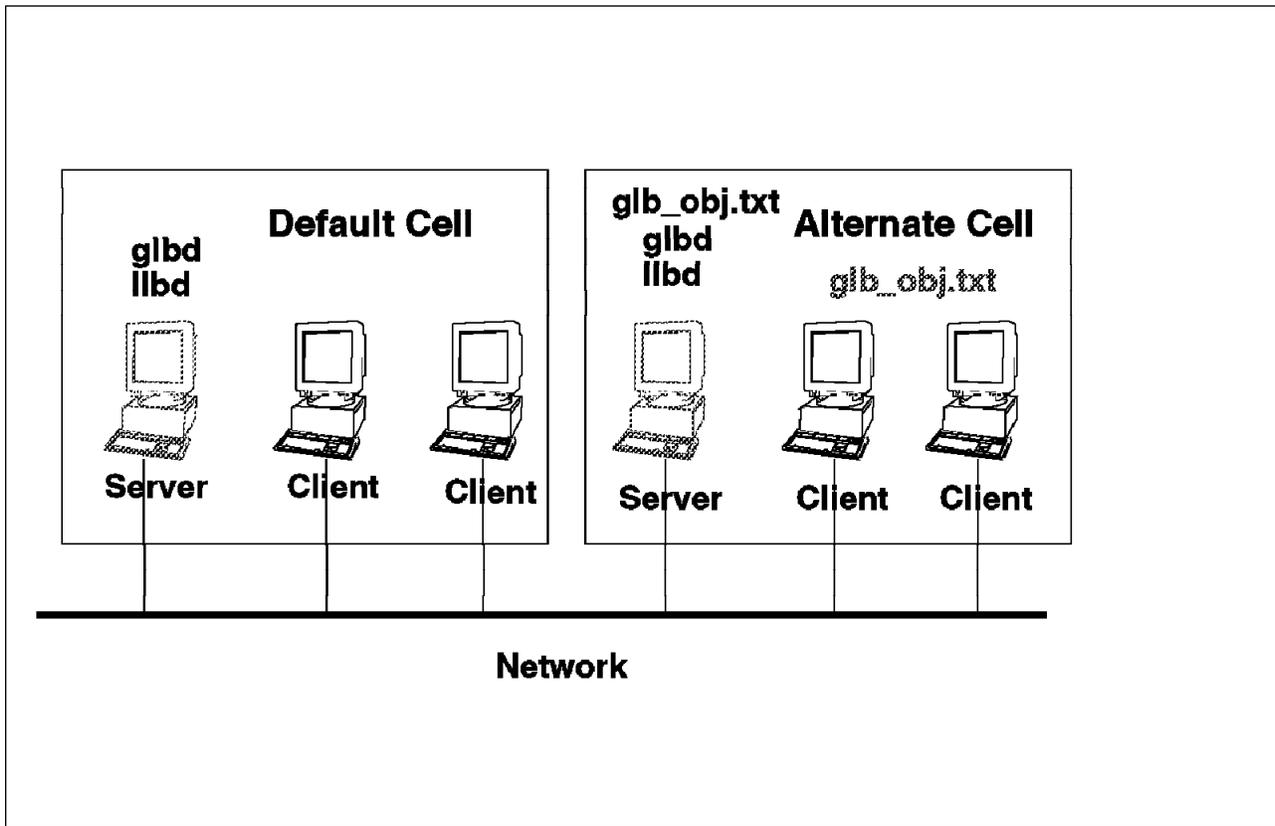
Figure 4. Default Cell versus Alternate Cell

2.5 General Migration Considerations

After a migration from AIX V3.2 to AIX V4.1, products that required a key in AIX V3.2 should be able to use the same key in AIX V4.1 since the CPU ID has not changed. This is true for both nodelocked license keys and for floating license keys.

Note: This is not true when upgrading a UP system to an SMP system. The CPU ID will not be the same; thus new license keys will be required.

If NetLS is used on your AIX V3.2 system to serve licenses, you should be able to still use NetLS after the migration to AIX V4.1 with the same keys.

If you choose to upgrade the NetLS license server to iFOR/LS, you can still use the same keys that were used for NetLS. You can actually even use the same license database; so there is no need to reinstall the keys after the migration.

2.6 iFOR/LS Packaging

The iFOR/LS runtime environment, also known as ARK (Administrator Runtime Kit) is shipped with AIX V4 in three filesets:

- iFOR/LS Libraries: bos.rte.ifor_ls
- iFOR/LS License System Client Utilities: bos.ifor_ls.client
- iFOR/LS License System Server: bos.ifor_ls.server

The first fileset belongs to the bos package. The two other filesets belong to the bos.ifor_ls package.

2.6.1.1 iFOR/LS Libraries

This fileset (bos.rte.ifor_ls) contains the Network Computing Kernel (NCK) shared libraries: libnck.a for NCS V1.1 and libshrnck.a for NCS V1.5.1. It also contains the iFOR/LS libnetls_shr.a shared library.

2.6.1.2 iFOR/LS License System Client Utilities

This fileset (bos.ifor_ls.client) contains the iFOR/LS administrative tools and utilities like `ls_admin`, `ls_tv`, `ls_rpt`, `ls_stat` and `ls_targetid`. It also contains SMIT screens and the X11R5 library stubs.

This fileset is needed to install and administer licenses and generate license usage reports.

2.6.1.3 The iFOR/LS License System Server

This fileset (bos.ifor_ls.server) includes the `netls_config` configuration shell script, the iFOR/LS daemon `netlsd` and the `ls_dpass` tool.

This fileset is needed if the machine is being configured as an iFOR/LS server. The `ls_dpass` tool is used to access and manipulate compound licenses.

2.7 How to Obtain License Keys

In order to get license keys for a product, you will need to fill out the IBM Product Registration Information Form before contacting the Registration Center. A template Information Form is supplied in the following file:

`/usr/lpp/<product_name>/README.password`

where `<product_name>` stands for the product name for which you need the iFOR/LS license key. For example, the C Set ++ for AIX product name would be xlC.

For more information on how to fill out this form, please refer to the redbook *A Holistic Approach to AIX V4.1 Migration, Planning Guide*, SC24-4651.

This form requires that you supply the iFOR/LS target ID of the node where you want to install the product licenses. You can do this by using the `ls_targetid` command or the `uname -m` command.

Note: In order to use the ls_targetid utility, you must at least install the iFOR/LS License System Client Utilities fileset (bos.ifor_ls.client) on the node where the licenses are going to be installed.

Run the `ls_targetid` command as follows:

```
#/usr/lib/netls/bin/ls_targetid

            PREFERRED iFOR/LS Target ID (CPU Planar ID)
            -------------------------------------------
                1866946


            ALTERNATE Target ID (Link Level Address)
            ----------------------------------------
                812317FA

#
```

The information returned is unique for each iFOR/LS license server node.

Similar information can be obtained with the `uname -m` command.

```
# uname -m
000186694600
```

To convert the `uname` output into the `ls_targetid` format, take off the leading zeros and the two trailing zeros. In this example, the target ID is 1866946.

2.8 How to Set Up a Nodelocked License

To install a nodelocked license on a machine for a specific product, do the following:

- Install the bos.ifor_ls.client fileset on the system. You do not need to install the bos.ifor_ls.server fileset.

- Set the system clock to the correct time; otherwise your license may not be recognized as active.

- Run the `ls_targetid` command in order to request the license key.

- If it does not already exist, create the file /usr/lib/netls/conf/nodelock.

- Change the nodelock file permissions to 644 so everybody can read it using the following command:

  `chmod 644 /usr/lib/netls/conf/nodelock`

- Edit the nodelock file using your favorite editor.

- Insert the nodelocked license as it appears in the printout (mailing or fax) you received from the Registration Center.

  The nodelocked license that you received should look similar to the following:

```
Vendor Password : agkrmbnthcsc4
built as
    vendor name : Common
    vendor id   : 5ce6f0617216.02.81.23.85.e3.00.00.00
    target      : 1866946

Product Passwords built as:
    name        : demo
    id          : 1926
    version     : 3.3
    license type : Node Locked

Product password for 0x1866946 server : wkseneaansmw6u6wjmsjp65t6w
built as
    start date  : 04/08/94
    duration    : 30
    expiration  : 05/07/94
    target type : IBM/AIX
    target      : 1866946
```

The license key is made up of the following fields:

 – Vendor ID   : 5ce6f0617216.02.81.23.85.e3.00.00.00
 – Product Password : wkseneaansmw6u6wjmsjp65t6w
 – Annotation   : ""
 – Version   : "3.3"

Note: The annotation field is not used in AIX Version 4.1.3, but must be inserted into the line you put into the nodelock file as "".

- Insert the following line into your nodelock file:

  `5ce6f0617216.02.81.23.85.e3.00.00.00 wkseneaansmw6u6wjmsjp65t6w "" "3.3"`

- Add a comment line above the license, so that if necessary, you can identify the license in the future.

  Note: Comment lines are identified by a # character in the first position.

Attention!

When editing the nodelock file, you should be aware of the following:

- Licenses are case sensitive, letters must be lower-case.
- Do not confuse the number one (1) with the lower-case letter L (l).
- Do not mistake the number zero (0) for an upper-case letter o (O).
- Do not attempt to substitute a single quote for a double quote in the license.
- Licenses cannot be split by a carriage return.

You will find below a sample nodelock file:

```
# Test Product Version 3.3. Valid 03/12/94 - 03/11/95
9ce6f0617216.02.81.23.85.e3.00.00.00 89pbxmywy8qh9mdu97u8yb586naa "" "3.3"

# Real Product Version 4.1. Annotation: "abc". Valid 04/08/94 - 06/01/94
90b82be84513.02.81.23.86.fb.00.00.00 26z8g3vnvdsn8b4u3n8bquspefmsnamj "abc" "4.1"

# Common "demo" Product Version 3.3  Valid 04/08/94 - 05/07/94
5ce6f0617216.02.81.23.85.e3.00.00.00 wkseneaansmw6u6wjmsjp65t6w "" "3.3"
```

Note that setting up nodelocked licenses can also be done through SMIT with the following path:

```
smitty
-> Software License Management
   -> Manage Nodelocked Licenses
```

2.9 How to Set Up Concurrent-Use Licenses

This is a description on how to install concurrent-use licenses (also called floating licenses) for a simple configuration made of one license server and several clients that can concurrently access the same product.

In this configuration, setting up floating licenses consists of the following steps:

- Install the iFOR/LS and prerequisite software on the license server.
- Configure NCS and iFOR/LS on the server.
- Start the NCS and iFOR/LS server daemons.
- Configure each client machine.

2.9.1 Installing iFOR/LS Software on the Server

The following iFOR/LS filesets are required to configure iFOR/LS on the floating license server:

- bos.ifor_ls.client
- bos.ifor_ls.server
- bos.net.ncs

This bos.net.ncs fileset will provide the following daemons:

- `llbd`, the NCS local location broker daemon
- `glbd`, the replicable global location broker daemon
- `nrglbd`, the non-replicable global location broker daemon

Note: Both bos.ifor_ls.client and bos.net.ncs can be installed individually. But if bos.ifor_ls.server is being installed, then the filesets bos.ifor_ls.client and bos.net.ncs will automatically be installed as corequisite filesets.

2.9.2 Configuring NCS and iFOR/LS on the Server

The `netls_config` script will allow you to configure an NCS cell and start NCS and iFORL/LS daemons.

This script will gather NCS configuration information and place it into a file called /usr/lib/netls/conf/netls_first_time. You can use the default cell or an alternate cell. For testing purposes, we recommend that you establish an alternate cell for your server. This will keep license requests for your test environment separate from license requests for other environments.

In order to configure the floating license server, do the following:

- Run the following shell script:

  `/usr/lib/netls/conf/netls_config`

- When prompted by the following questions, give the following answers:

Q1. Do you want the `llbd` started automatically when the
machine boots?
>> answer **y**

Q2. Do you want `netlsd` started automatically when the
machine boots?
>> answer **y**

Q3. (You might be asked this question)
There already exists an initialized database for the
`glbd`. Do you wish to use that database when starting
the `glbd` daemon?
>> answer **n**

Note: You will be shown a listing of all the cells that your machine can
identify. If you are setting up your machine in a new alternate cell, none of this
information is relevant to you. If you intend to join an alternate cell that has
already been established, note the cell number when it is displayed on the
screen. It should also be pointed out that the cell numbers may be assigned
differently each time `netls_config` is run.

Q4. You have four options:

1. Continue with installation without choosing a Cell Name.
2. Use the default for the system Cell Name.
3. Create a new alternate cell for the system Cell Name.
4. Choose an existing alternate cell for the system Cell Name.

Attention!

If you continue the install without choosing a cell name [1], then iFOR/LS
will not function. If you do not understand these options, enter [1] to
continue, and consult your system administrator or the document *Managing
NCS Software*.

Please indicate your choice [1,2,3,4]:

– Answer **1** if you do not know what configuration you want, and you just
want to quit. You will have to re-run `netls_config` later.

– Answer **2** only if your site administrator tells you it is ok or if you are on a
stand-alone machine.

– Answer **3** if you are going to run your server in a brand new alternate cell.

– Answer **4** if you are going to join your server to a cell that has already been
established. You will then be prompted for the number of the alternate cell
you want to join. Enter that number now.

• You are now done with the `netls_config` script. It has now created a
secondary shell script called /usr/lib/netls/conf/netls_first_time.

• Run the `netls_first_time` script. This will start the daemons needed for
serving license requests. This script only needs to be run once.

After the first time, the daemons can simply be started with the `startsrc`
command if you did not select them to be started automatically after each
machine boot. If you do so, you must start them in the correct order: `llbd`,
then `glbd`, then `netlsd`.

The exact commands are:

```
# startsrc -s llbd
# startsrc -s glbd
# startsrc -s netlsd
```

2.9.3 Installing Licenses on the Server

Now that your license server is up and running, install your floating licenses. You can do this with either the command line or with the graphical user interface provided by iFOR/LS. In both cases, the data entered must be EXACTLY the same as it appears on the printout you received from the Registration Center. Cutting and pasting in X-windows is the easiest way of avoiding problems.

The floating license that you received should look similar to the following:

```
Vendor Password agkrmbnthcsc4
built as
   vendor name : Common
   vendor id   : 5ce6f0617216.02.81.23.85.e3.00.00.00
   target      : 1866946
ls_admin -a -v "Common" 5ce6f0617216.02.81.23.85.e3.00.00.00 agkrmbnthcsc4

Product Passwords built as
   name         : demo
   id           : 1926
   version      : 3.3
   license type : Concurrent Access

Product password for 0x1500910 server : s8yewwjtbfh68jf4wewbvuf3jiaaa
built as
   start date      : 04/08/94
   duration        : 90
   expiration      : 07/06/94
   number          : 1
   target type     : IBM/AIX
   target          : 1866946
ls_admin -a -p "Common" "demo" s8yewwjtbfh68jf4wewbvuf3jiaaa "3.3"
```

2.9.3.1 Installing Licenses Using the Command-Line Interface

To install licenses using the command line, simply type in (or cut and paste to the command line) the two lines above that start with `ls_admin` (your path must include `/usr/lib/netls/bin`). The first line adds the vendor to your license database. The second line adds the specific product licenses for that vendor in your license database. Once the first line has been entered (in other words, once a vendor has been added), it does not have to be entered again for other product licenses provided by the same vendor. In this example, the two lines would be:

```
# ls_admin -a -v "Common" 5ce6f0617216.02.81.23.85.e3.00.00.00 agkrmbnthcsc4
# ls_admin -a -p "Common" "demo" s8yewwjtbfh68jf4wewbvuf3jiaaa "3.3"
```

2.9.3.2 Installing Licenses Using the Graphical Interface

To install licenses using the graphical user interface, run the following command with no flags:

```
# /usr/lib/netls/bin/ls_admin
```

Figure 5 shows the iFOR/LS graphical user interface.
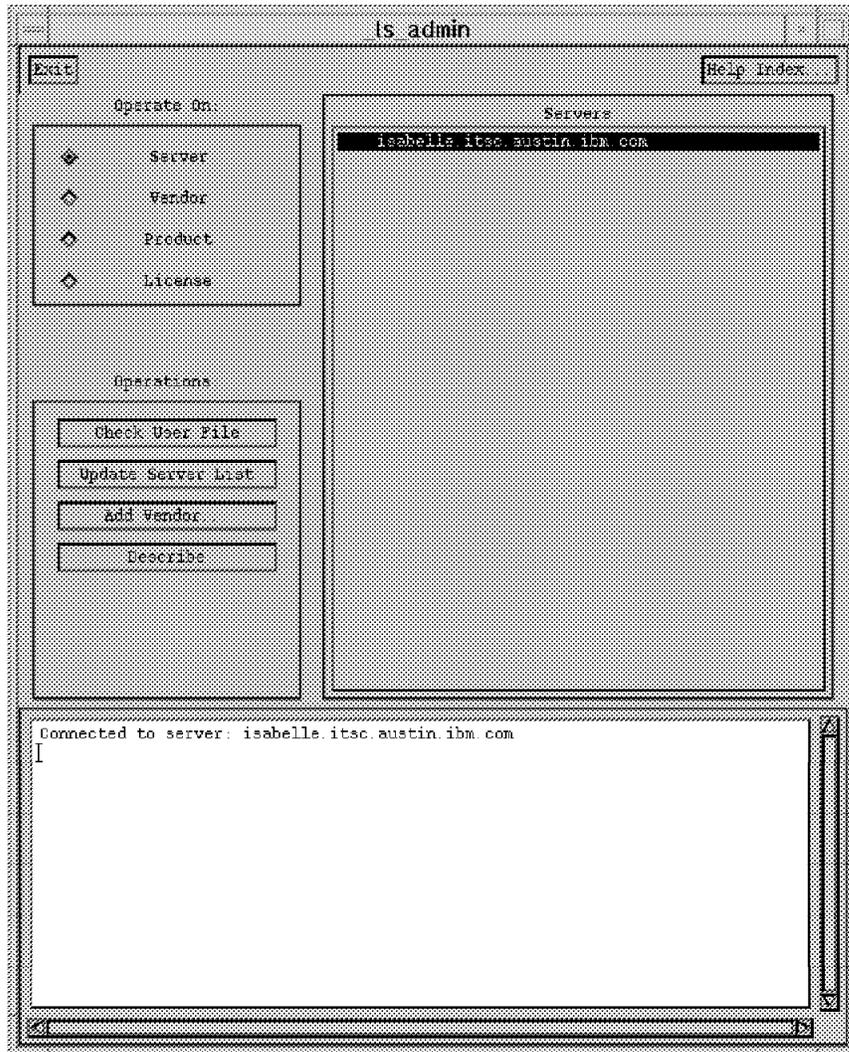
Figure 5. iFOR/LS Graphical User Interface

To install concurrent-use licenses, do the following:

1. In the Servers display box, click on the server where you will install the license.

2. In the Operations box, click on **Add Vendor**

3. In the pop-up window for ADD VENDOR, click on the **Vendor Name** button. A window for Vendor Name will pop up. Enter the vendor name, and click on the **OK** button. In the example above, the vendor name would be:

   Common

4. A window should pop up for Vendor ID. If it does not, then just click on the button in Add Vendor for **Vendor ID.** Enter the vendor ID, and click on the **OK** button. In the example above, the vendor ID would be:

```
5ce6f0617216.02.81.23.85.e3.00.00.00
```

5. A window should pop up for Vendor Password. Enter the vendor password, and click on the **OK** button. In the example above, the vendor password would be:

```
agkrmbnthcsc4
```

6. Click on **done** in the Add Vendor window. Your vendor should be added.

7. Now add the product license for the newly added vendor. Click on the **Vendor** button in the Operate On box.

8. In the Vendors display box, click on the vendor whose product you will be adding.

9. In the Operations box, click on the button for **Add Product.**

10. In the pop-up window for Add Product, click on the **Product Name** button. A window for Product Name will pop up. Enter the product name, and click on the **OK** button. In the example above, the product name would be:

```
demo
```

11. A window should pop up for Product Version. Enter the product version, and click on the **OK** button. In the example above, the product version would be:

```
3.3
```

12. A window should pop up for License Annotation. If a license annotation was supplied with your license, enter it now, and click on the **OK** button. If no annotation was supplied, then leave the field blank, and just click on **OK.** In the example above, no annotation was supplied.

13. A window should pop up for License Password. Enter the license password, and click on the **OK** button. In the example above, the license password would be:

```
s8yewwjtbfh68jf4wewbvuf3jiaaa
```

14. Click on **done** in the Add Product window. Your license password should be added.

To add a product for an already added vendor, perform the following steps:

1. Click on the **Server** button in the Operate On box. Then click on the server you will add the license to in the Servers display box.
2. Click on the **Vendor** button in the Operate On box. Then click on the vendor you will add the product for in the Vendors display box.
3. Click on the **Add Product** button in the Operations box.
4. Add the product information as you normally would.

To add a license for an already added product, perform the following steps:

1. Click on the **Server** button in the Operate On box. Then click on the server you will add the license to in the Servers display box.
2. Click on the **Vendor** button in the Operate On box. Then click on the vendor you will add the product for in the Vendors display box.
3. Click on the **Product** button in the Operate On box. Then click on the product you will add the license for in the Products display box.
4. Click on the **Add Licenses** button in the Operations box.
5. An Add Licence window will pop up. If your licenses has an annotation, click on the button for **license annotation**, and specify the licenses annotation that was supplied. Add the license password by clicking on the **License Password**

button in the Add Licence window, and specify the license password that was supplied.

6. Click on the **Done** button in the Add Licence window when you have added the license information.

7. Your license should be added.

Note: The same operations can be performed by using the following SMIT path:

```
smitty
-> Software License Management
   -> Manage License Servers and License Databases
```

2.9.4 Configuring the Client Machines

The final step consists in configuring the client machines that need to use the licensed product.

- Install the following filesets on the client machine

  – bos.rte.ifor_ls

  – bos.ifor_ls.client

- Copy or FTP the file /etc/ncs/glb_obj.txt from the license server to the client machines. This file simply contains the identifier of the alternate cell.

  This is an example of the content of the /etc/ncs/glb_obj.txt file:

  ```
  90b82be849c7.02.81.23.86.fb.00.00.00
  ```

- If the client machine is on a different subnet than the server machine, then it will also need the file /etc/ncs/glb_site.txt. The content of this file will identify the license server machine.

  For example, if the server machine is `my_server.my_net.my_company`, then the glb_site.txt file will contain the line:

  ```
  ip:my_server.my_net.my_company
  ```

Note: The glb_site.txt file can also specify the IP network address of the server machine. For example, if the IP address of the host machine is `111.22.333.444`, then the glb_site.txt file file will contain the line:

```
ip:#111.22.333.444
```

2.10 Verifying iFOR/LS Server Installation

There are a few commands that can be used to check and test an iFOR/LS configuration. These commands are located in the directory /usr/lib/netls/bin.

The commands are:

- **ls_tv**:

  This command is a license server test and verification tool. It verifies that the license server is working.

- **ls_stat**:

  The `ls_stat` command provides status information on network licenses (that is, all license types except nodelocked). End users, as well as system

administrators, may find the `ls_stat` command useful for finding out the status of licenses.

- **ls_rpt**:

  This command generates reports on license server events. There is no graphic interface for this command.

2.10.1 Checking the iFOR/LS Installation

In order to verify that your license server is working fine, you can use the `ls_tv` command. If you receive one of the error messages listed below, this means that something went wrong in the installation.

These are a subset of the `ls_tv` error messages:

- **netls_request_license**: There is a communication failure due to the NCS/RPC runtime.
- **netls_no_svrs_found**: No servers are available for this vendor.
- **netls_lic_not_fnd**: The license was not found in the database.
- **netls_not_authorized**: The user is not authorized to use this product.
- **netls_bad_timestamp**: Time disparity is too large.

Note: A complete list of `ls_tv` error messages can be found in the *AIX Version 4.1 iFOR/LS System Management Guide*, SC23-2665.

If you can run `ls_tv` successfully but are still having a problem with a license product, the problem is probably with the licenses or possibly with the product itself. In this case, talk to the vendor of the licensed software product.

If you cannot run `ls_tv` successfully and receive an error that is not listed below, it means there is a problem with the software on which the license server is layered (for example TCP/IP or NCS), or it may be a hardware problem.

In this case, we recommend that you do the following:

- **Check the netlsd daemon configuration:**

  - Does the file /etc/inittab contain the /etc/rc.netls startup procedure?

  - Is the /etc/rc.netls file in a good shape?

  - Are the database files of the `netlsd` still in place? Check the files /usr/lib/netls/conf/lic_db and /usr/lib/netls/conf/cur_db.

  - NCS and iFOR/LS are based on the User Datagram Protocol (UDP). In a very highly loaded network, UDP connections may receive timeouts before data is delivered. There is nothing wrong with this behavior. Reduce the total network load.

- **Check NCS:**

  - Does the file /etc/inittab contains the /etc/rc.ncs startup procedure?

  - Is the /etc/rc.ncs file in a good shape? The first daemon started there must be `llbd`. The second daemon should be the `glbd`.

  - Is the `llbd` daemon running?

  - Is the `glbd` daemon running?

  - Are the system clocks of the NCS servers (machines running `glbd` in the same NCS cell) still in sync? This is mandatory to make NCS work. Use

the `setclock` command to sync all systems for a short-term solution. It is recommended to implement external time providers and a distributed time service on the network.

  – Is the /etc/ncs/glb_obj.txt file the same on all NCS cell member hosts?

  – Is the /etc/ncs/glb_site.txt file (if needed) still pointing to a valid Global Location Broker (GLB) host?

  – Do the GLB database files still exist? Check particularly for the existence of the files /etc/ncs/glb.e and /etc/ncs/glb.p.

  – Was `llbd` able to create its temporary file /tmp/llbdbase.dat?

  – Use the `ncs_test.sh` script to test setup and runtime parameters.

    Note: This script will be inserted in the section called 2.10.2, "Testing the NCS Configuration for iFOR/LS."

- **Check TCP/IP:**

  – Has someone changed IP addresses or network interfaces ? The Global Location Broker database may not reflect the changes someone has done to the real network. Check the NCS config files, and restart the NCS and iFOR/LS daemons.

  – Can you reach other systems in the network?

  – Is your routing setup definition valid? The `netstat` command shows the local definition. To see the actual hubs, use the `traceroute` command.

  – Is the name resolution working? Name resolution is very often the reason for long startup times or many sorts of problems in large networks. Use the Domain Name System (DNS), and spend some time developing a good layout.

  – Is the Maximum Transmission Unit (MTU) size equal on all hosts?

  – Is the token-ring speed equal on all hosts?

- **Check the hardware:**

  – Has someone changed the CPU planar of your iFOR/LS server? The licenses on the iFOR/LS server are tied to the CPU ID, which changes after a CPU planar swap.

  – Are the cables still where they should be? Check the cables.

  – Have you reached the Ethernet length limitations on your LAN ? Most LANs are growing all the time, but there are limits. Be aware of them.

  – Has someone enabled a security feature on a router? Some routers allow enabling of security features. It is possible to block certain TCP/IP ports. The `llbd` program is running on port 135. The `glbd` and `netlsd` programs are using runtime-assigned ports whose port numbers are greater than 1024.

2.10.2 Testing the NCS Configuration for iFOR/LS

The `ncs_test.sh` shell script can be used to ensure all the definitions in the NCS environment are set up properly. This shell script uses the `lsof` command, if installed, to find out the port numbers on which the `llbd`, `glbd`, and `netlsd` daemons are communicating.

The `lsof` command is public-domain software and can be found on the internet at `vic.cc.purdue.edu` (`128.210.15.16`). The identification of the runtime-assigned `glbd` and `netlsd` ports is not essential. Therefore, the shell script will not break if the `lsof` software is not installed.

After the UDP ports have been identified, the `ncs_test.sh` shell script traces the well-known `llbd` UDP port (135) and collects information on the traffic caused by the `lb_admin` command. The formatted report is displayed after it is processed by the `ipreport` command.

```
┌─ Attention! ──────────────────────────────────────────────────────┐
│                                                                    │
│  This shell script should be used for reference only. It has not been submitted to │
│  any formal test and is distributed AS IS.                         │
│                                                                    │
└────────────────────────────────────────────────────────────────────┘
```

```
#!/bin/ksh
# @(#) ncstest.sh NetLS/NCS 1.5.1 verify/debug script 94/03/15
#===================================================================#
#    Authors :  M. Crisanto / L. Denefleh / F. Kraemer            #
#-------------------------------------------------------------------#
LSOF="/usr/local/bin/lsof"          #  path to lsof command
LBADMIN="/usr/lib/ncs/bin/lb_admin" #  path to lb_admin command
AWK="/usr/bin/awk"                  #  path to awk  command
CUT="/usr/bin/cut"                  #  path to cut  command
SED="/usr/bin/sed"                  #  path to sed  command
MYPID="$$"                          #
LPORT="135"                         #  llbd runs here
IPTRACE="1"                         #  start iptrace (1=yes/0=no)
IPTRFILE="/tmp/iptrace.$MYPID"      #  filename of iptrace output
#-------------------------------------------------------------------#
#  Prepare some staff before doing real work.                      #
#-------------------------------------------------------------------#
if [ "$(/usr/bin/whoami)" != "root" ]; then
  echo "\n\tYou must be root to run this script."
  exit -1
fi
TMP_FREE=$(/usr/bin/df "/tmp" | $AWK '$3 ~ /[0-9]/{print $3}')
if [ "${TMP_FREE}" -lt 1000 -a "${IPTRACE}" = "1" ]; then
   echo "\n\tThere is not enough room in your /tmp directory."
   echo "\tYou need 1000 KB free, and you have only $TMP_FREE KB free.\n"
   exit -1
fi
if [ ! -x ${LBADMIN} ]; then
   echo "\n\tCan not find the ${LBADMIN} command on the system."
   echo "\tPlease verify the NetLS installation.\n"
   exit -1
fi
#-------------------------------------------------------------------#
#  Ok all checks are done we can take off.                         #
#-------------------------------------------------------------------#
PROG=$(basename $0)
HOST=$(hostname -s)
TODAY=$(date +%H:%M:%S)
echo "\n\t$PROG started from $LOGNAME@$HOST on $TERM at $TODAY.\n"
#-------------------------------------------------------------------#
# Use lsof to find the portnumber of llbd, glbd and netlsd daemon #
```

```
                    #----------------------------------------------------------------#
                    if [ -x $LSOF ]; then
                       LRT=$($LSOF -i"UDP" | $AWK '{if ($1 == "llbd")   print $9}' | $CUT -c3-7)
                       GRT=$($LSOF -i"UDP" | $AWK '{if ($1 == "glbd")   print $9}' | $CUT -c3-7)
                       NRT=$($LSOF -i"UDP" | $AWK '{if ($1 == "netlsd") print $9}' | $CUT -c3-7)
                    #--------
                       LPORT=$(echo $LRT | $SED 's/\\n/ /g')   #  Format the staff
                       GPORT=$(echo $GRT | $SED 's/\\n/ /g')   #
                       NPORT=$(echo $NRT | $SED 's/\\n/ /g')   #
                    #--------
                       if [ "$LPORT" = "" ]; then              #  Print info
                          echo "\n\t** No Local Location Broker is running on local system **"
                       else
                          echo "\n\tLLB is using UDP port(s)   := $LPORT"
                       fi
                    #--------
                       if [ "$GPORT" = "" ]; then              #  Print info
                          echo "\n\t** No Global Location Broker is running on local system **"
                       else
                          echo "\n\tGLB is using UDP port(s)   := $GPORT"
                       fi
                    #--------
                       if [ "$LPORT" = "" ]; then              #  Print info
                          echo "\n\t** No NetLS daemon is running on local system **"
                       else
                          echo "\n\tNetLS is using UDP port(s) := $NPORT\n"
                       fi
                    else
                       echo "\n\tlsof - List_of_Open_Files is not installed on your system."
                       echo "\tthe tool is a public domain program and can be found on"
                       echo "\tvic.cc.purdue.edu (128.210.15.16), Vic Abell is the author.\n"
                    fi
                    #----------------------------------------------------------------#
                    #  Fire up an IP trace on the llbd UDP (135)                     #
                    #----------------------------------------------------------------#
                    if [ $IPTRACE = "1" ]; then
                       /usr/bin/rm -f ${IPTRFILE} 2>/dev/null
                       /usr/bin/iptrace -P "UDP" -p "$LPORT" ${IPTRFILE}
                       sleep 3
                    fi
                    #----------------------------------------------------------------#
                    #  Use lb_admin command to hear on the llbd UDP port (135)       #
                    #----------------------------------------------------------------#
                    cat <<EOF | ${LBADMIN}
                    set_timeout long
                    set_timeout
                    use_broker local
                    use_broker
                    lookup
                    use_broker global
                    lookup
                    quit
                    EOF
                    #----------------------------------------------------------------#
                    #  Stop the IP trace and format its output                       #
                    #----------------------------------------------------------------#
                    if [ $IPTRACE = "1" ]; then
                       Target=$(ps -e | grep "iptrace")
```

```
        echo "\n\tKilling $Target with signal 1"
        /usr/bin/kill 1 $(echo $Target | cut -f1 -d" ")
        echo "\n\tFormating iptrace output via ipreport.....please wait\n"
        sleep 3
        /usr/bin/ipreport -r ${IPTRFILE}
        /usr/bin/rm -f ${IPTRFILE}
fi
#-----------------------------------------------------------------#
#  We are done.....hope you had fun.                              #
#-----------------------------------------------------------------#
exit 0
```

2.11 Installing AIX Floating User Licenses

Before the migration to AIX V4.1, you might have ordered AIX Networked User
Licenses, also referred to as AIX Floating User Licenses. This type of license
allows the system administrator to define AIX user licenses on an iFOR/LS server,
that are not restricted to a specific machine but that can float on any machine
(node) in the iFOR/LS cell.

When a user tries to log into one of the nodes within the iFOR/LS cell, if no fixed
license is available on this node, then the user can use one of the available floating
licenses defined on the iFOR/LS server. If no AIX floating license is available, the
user will not be able to log in.

The way to set up AIX Floating User Licenses is the following:

- If it does not already exist, set up your iFOR/LS server. The procedure is the
  same as described in 2.9, "How to Set Up Concurrent-Use Licenses" on
  page 79.

- Tell your iFOR/LS server that you want to use AIX User Floating Licenses by
  entering the following:

  ```
  # smitty
  -> System Environments
     -> Change / Show Number of Licensed Users
  ```

  Set the FLOATING licensing option to on.

  Or, enter the following command:

  ```
  # chlicense -f on
  ```

  This will activate the monitord daemon and add an entry into the /etc/inittab file
  for the monitord daemon.

- On the iFOR/LS server, register your AIX User Floating Licenses using the
  following SMIT path:

  ```
  # smitty
  -> System Environments
     -> Manage AIX Floating User Licenses for this Server
        -> Register Floating User Licences
  ```

  Or use the following SMIT fastpath:

  ```
  # smitty install_float
  ```

  Add the number of AIX Floating User Licenses you purchased.

- Check that your iFOR/LS client nodes participating in the iFOR/LS cell have the right glb_obj.txt file and the right glb_site.txt file (if necessary).

- Check on your client nodes that you can see the iFOR/LS server by using the `ls_tv` command.

- Activate the `monitord` daemon on all your iFOR/LS client systems by entering on each one the following command:

  `# chlicense -f on`

  Note: Since the `monitord` daemon and the `chlicense` command is part of the bos.sysmgt.loginlic fileset, this fileset must be installed on your client nodes. This fileset is only shipped with the AIX Version 4.1 for Servers. It is not part of the AIX Version V4.1 for Clients.

This will normally allow you to use AIX Floating User Licenses within your iFOR/LS cell.

Chapter 3. How it Works

A migration installation starts by following the same boot process as other installation types, however after the initial system boot, a lot of specialized processing is performed to delete old software and to migrate configuration files where possible. This chapter details the flow of a migration installation, from the initial system boot from the installation media, through the migration of BOS and LPPs, to the first system reboots and device definition.

An understanding of this section is *not required* to perform a migration install. In most cases, the steps detailed below will occur automatically without the knowledge of the user. By answering a few simple questions, the user can happily and successfully migrate their system. However, if the user has a particular requirement to vary the migration process or if the migration process fails for some reason the information in this section will provide the background knowledge required to successfully interrupt the migration at a particular point or recover from the error.

Attention!

While reading this chapter it is important to remember that this information is based on the AIX Version 4.1.4 installation process and shell scripts. The procedures may change with other operating system levels.

Also, many of the methods mentioned in this chapter for interrupting the installation scripts are intended for use by IBM development staff and are not supported for use by customers. While they may be useful under certain circumstances, any use of these exits is at your own risk.

In this chapter we will omit a lot of the specific details of Network Installation Management (NIM) Installation. In many ways NIM is sufficiently different to demand its own section. NIM installation and migration are detailed in *A Holistic Approach to AIX V4.1 Migration, Volume 2*

3.1 The Boot Process

The stages of the boot and installation process are shown in Figure 6 on page 92.

The process starts when the system reads the boot record from the installation device. The boot record consists of three components:

- Decompression code

- Compressed UNIX kernel

- Compressed boot file system (or RAM file system)

The decompression code is used to decompress the rest of the boot record. The compressed boot file system is also known as a RAM file system, since it will be decompressed into memory and used directly from there, rather than being used from disk or other media. As soon as the kernel is loaded, it is executed. As with most UNIX implementations, the kernel will immediately look to execute the init program which has the task of starting any other required programs. At this stage of the boot process, the boot file system contains a program `/etc/init`. This is not
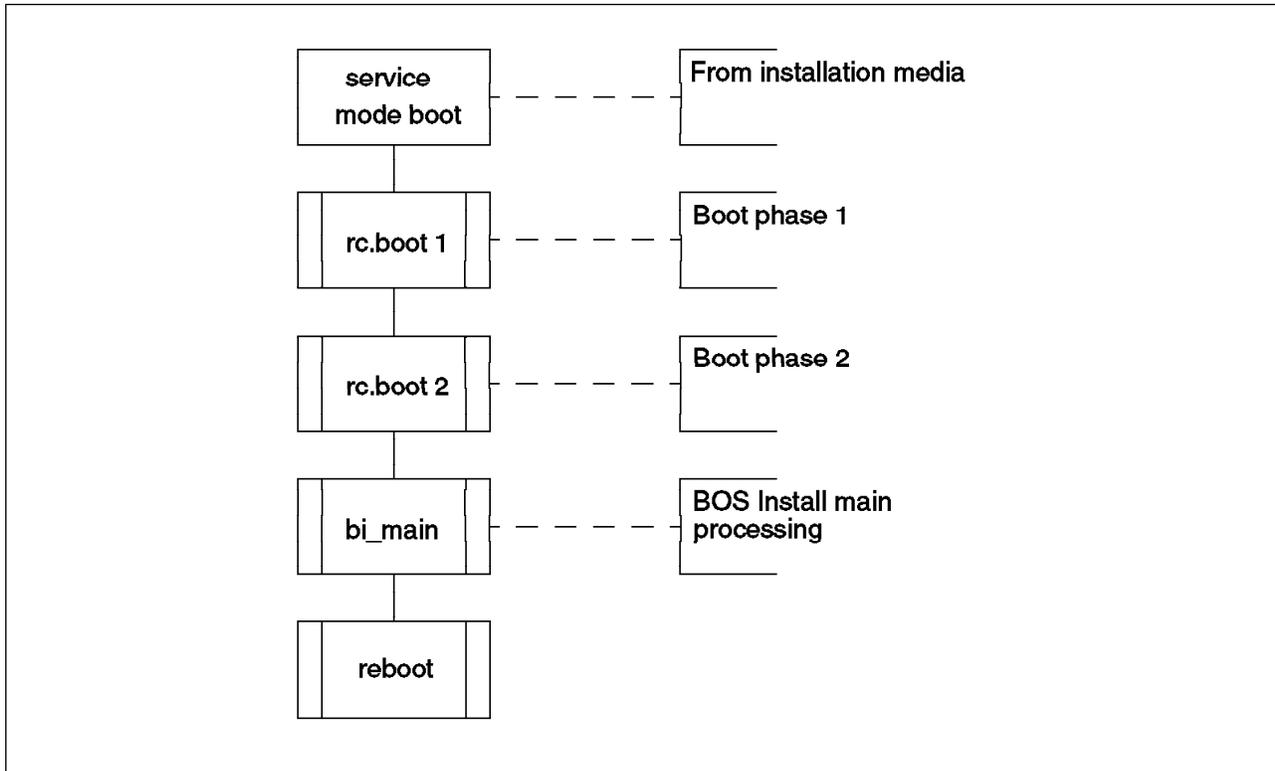
Figure 6. Boot and Installation Process

the full `init` program used under normal operation. It is a small shell program that calls the boot programs, and provides some recovery in case of errors. This init program is located in `/usr/lib/boot/ssh` on an AIX V4.1 system.

The AIX boot process is divided into three phases and is mostly controlled by the shell script `/sbin/rc.boot`. The init (ssh) program calls the shell script `/sbin/rc.boot` to execute both the first and second phases of the boot process. The boot phase is passed to the script as a parameter. Boot phase three is called by the real init program once the operating system is running from the rootvg volumes. This will only be called in the case of a hard disk or network boot and is not relevant to our consideration of the migration process.

The first two boot phases are detailed below.

3.1.1 LED Display Codes

Throughout this chapter, LED display codes are shown on the diagrams. These codes are important as they allow the user identify the location in the boot or installation process that is being executed at that time. However, in some of the newer RISC Systems that use the PCI bus, such as the model 43P, there is no LED display. In this case, it can be difficult for the user to identify the progress. There are two clues that give a rough indication of the state:

- As the PCI bus systems perform their Power-On Self-Test (POST), the system displays a graphic logo on a white background with series of icons representing the system components. Just before handing control to the boot kernel and entering boot phase one, these icons will be cleared.

- When performing phase two or three of device configuration during boot phase one, the logo screen is removed and replaced by a black screen with a green

cursor. This occurs when the graphics adapter is configured by the operating system.

3.2 Boot Phase One

Boot phase one is invoked by calling the `rc.boot` script with the parameter 1. Its major purposes are to:

- Determine the system architecture and configuration

- Detect and configure the available devices

- To access the system installation files

    (In one case this is done in phase two)

The major steps of this phase are shown in Figure 7 on page 94. A large part of the work involves device configuration using the `cfgmgr` command. This is explained in more detail in 3.7, "Device Configuration" on page 156. Note that the device configuration process also refers to phases, however these phases do not correlate with the boot phases.

The diagram shows the numbers that will appear on the LED display panel at each stage of the installation process. In addition, while `cfgmgr` runs, many other numbers will be displayed to indicate the specific devices that are being configured.

It is important to be aware that during boot phase one the system is operating using only the RAM file system and a small subset of operating system files. Often during the boot process, files or libraries that are no longer required will be removed before accessing more files. This reduces the size of the RAM file system required, and thus the overall memory size required to boot AIX.

Since the introduction of Version 4.1, AIX has supported the Rock Ridge format for CD-ROM drives. This has the advantage that the installation CD can be mounted as a file system. In this case it is not necessary to restore the installation files from the CD—the CD is simply mounted and some links are created from the RAM file system to the appropriate directories on the CD. In the case of a network installation using the Network Installation Management (NIM) a similar procedure is used. The installation files are NFS mounted across the network from the NIM installation server and links are created from the RAM file system to the appropriate directories.

The network boot differs from the CD-ROM or tape boot in several ways. As well as having to perform more sophisticated device configuration in order to access the network, it also has additional opportunities to tailor the boot and installation process. As part of boot phase one, the configuration script `<boot image name>.info` (where <boot image name> is the name of the boot image used) is copied from the server using tftp. This script is executed, and sets several environmental variables related to the NIM configuration, including the name of a NIM boot configuration file. By default this is called `/usr/lib/boot/network/rc.bos_inst`. This file is copied from the NFS mounted server directories into the RAM file system and executed. This shell script continues the process of accessing the appropriate installation files and directories.

determine boot device

tape

network
600

CD

600
determine network information

600
cfgmgr -f -v
Phase 1 - system and LVM

510
cfgmgr -f -v
Phase 1 - system and LVM

510
cfgmgr -f -v
Phase 1 - system and LVM

606
configure network adapter

608
tftp and execute client.info file

510
cfgmgr -p3 -v
Phase 3 - all other devices

517
mount CD

610
NFS mount /usr/SPOT/ filesystem

612
copy & execute configuration script

512
set tape blocksize to 512

512
link install directories

610
NFS mount NIM filesystems

622
link required device entries

512
remove unneeded files

510
cfgmgr -p3 -v
Phase 3 - all other devices

622
cfgmgr -f -v
Phase 1 - system and LVM

622
cfgmgr -s -v
Phase 2 - all other devices

511

512

exit

622

Figure 7. Boot Phase One. Bold numbers show LED display values.

3.3 Boot Phase Two

Calling the script `bi_main 2` invokes the second phase of the boot process. As shown in Figure 8, the second phase consists mainly of calling the `bi_main` shell script. This script performs most of the installation and migration processing, and is explained in 3.4, "Installation Processing" below.
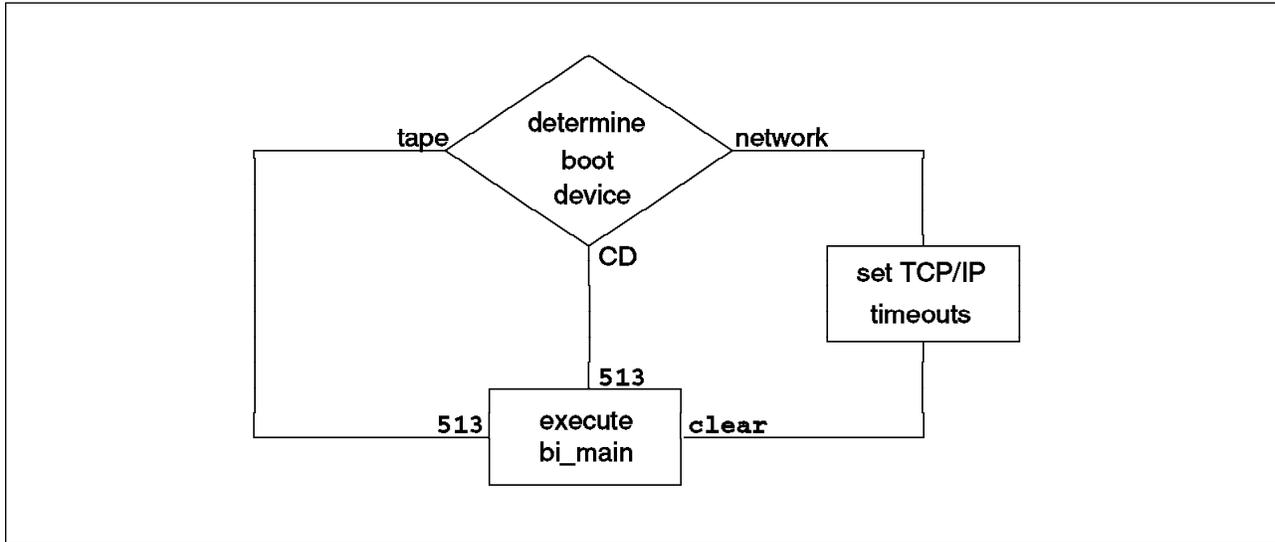
Figure 8. Boot Phase Two.   Bold numbers show LED display values.

3.4 Installation Processing

The shell script `/usr/lpp/bosinst/bi_main` is the heart of the Base Operating System (BOS) installation. It takes the system from boot phase two, through the selection of a console, the installation menus and the actual installation or migration of the operating system, to the eventual system reboot. Thus the script is long (around 6500 lines!) and complex. We have split this chapter and the flowchart showing the major functions of `bi_main` into several sections. The major sections of `bi_main` are shown in Figure 9 on page 96.

3.4.1 Error Handling in bi_main

Before starting the discussion of the `bi_main` script it is worthwhile establishing the methods used for handling any errors that are encountered during the script. There are four different methods that are used depending upon the stage of `bi_main` and the resources available at that stage.

3.4.1.1 loopled
Most errors that occur very early in the installation process cannot be recovered through immediate action by the user. This includes errors such as being unable to locate the installation device. At this stage of the installation process, the console has not yet been selected and configured. For this reason, the only available method of communication with the user is the LED display on the system front panel. The `loopled` function is called to display a value on the LED display, then to start an infinite loop—effectively halting the system. This method is also used in some cases of errors that occur during an un-prompted installation when the console is not defined.

```
          ╭─────────────╮
          │    start    │
          ╰──────┬──────╯
          ┌──────┴──────┐
          ││ load data  │
          ││   files    │
          └──────┬──────┘
        ┌────────┴────────┐
        ││   determine    │
        ││installation method│
        ││ and target disks │
        └────────┬────────┘
          ┌──────┴──────┐
          ││  validate  │
          ││bosinst.data│
          ││   values   │
          └──────┬──────┘
          ┌──────┴──────┐
          ││  validate  │
          ││ image.data │
          ││   values   │
          └──────┬──────┘
          ┌──────┴──────┐
          ││    user    │
          ││   menus    │
          └──────┬──────┘
          ┌──────┴──────┐         ┌──────────────────────┐
          ││  prepare   │         │Includes BOS migration │
          ││  target    │ ─ ─ ─ ─ │pre-installation processing│
          ││   disks    │         │and migration menus    │
          └──────┬──────┘         └──────────────────────┘
          ┌──────┴──────┐
          ││  restore   │
          ││ BOS files  │
          └──────┬──────┘
          ┌──────┴──────┐
          ││ initialize │
          ││    disk    │
          ││environment │
          └──────┬──────┘
          ┌──────┴──────┐
          ││   change   │
          ││   mounts   │
          └──────┬──────┘
          ┌──────┴──────┐         ┌──────────────────────┐
          ││    post    │         │Includes BOS migration │
          ││installation│ ─ ─ ─ ─ │post-installation and  │
          ││ processing │         │LPP migration          │
          └──────┬──────┘         └──────────────────────┘
          ┌──────┴──────┐
          ││    run     │
          ││customization│
          └──────┬──────┘
          ╭──────┴──────╮
          │    exit     │
          ╰─────────────╯
```

Figure 9. Installation Processing in bi_main.

In the PCI bus RISC Systems that do not have LED codes, there is no way to be notified of these errors and faults in this stage can be difficult to diagnose. The two clues described on 3.1.1, "LED Display Codes" on page 92 can be of some assistance.

3.4.1.2 ERROR_EXIT
The error exit provides the user with an opportunity to provide their own tailored error handling routine. The ERROR_EXIT routine can be specified in the bosinst.data file which is read early in the installation process. (The bosinst.data file is explained in 3.8.1, "bosinst.data file" on page 160.) With most errors that occur, the error exit routine will be called if one is defined. If none is defined, or if the error routine returns control to `bi_main`, one of the standard methods below will be used.

3.4.1.3 ERRORTEXT and ERRORNUM
This method of error handling is used mostly when reading and processing the various input data files, up until the time when the first user menu screen is displayed. During this stage if an error is encountered with a particular value in one of the input data files we would like to continue processing the rest of the files before acting upon the error. For this reason no immediate action is taken, however the script will set variables ERRORTEXT and ERRORNUM to contain a description and identifier for the error if they are not already set. This ensures that the first error will be reported. The PROMPT variable in the bosinst.data file will be set to **yes**, regardless of the user's original setting, ensure that the user menus will be displayed. The error message will not be displayed until the installation menus are started. The error message will then be displayed in the corner of the screen.

3.4.1.4 BI_Error
The function `BI_Error` is used to report any errors that occur after the user menus. The function will stop any status displays that may be in progress before displaying any errors. If the system is being installed from a NIM server, the error message will be sent to the NIM server. If the error exit is defined, it will then be called. If there is no error exit routine defined and if no console is defined or an unprompted installation is specified, the only option is to call `loopled`. Otherwise, the function will call the program `/usr/lpp/bosinst/berror` which will display an error message then prompt the user to either continue, or to perform system maintenance and then continue.

Note that not every call to `berror` indicates an error. It is also used as a general messaging routine to permit the display of international language supported messages. Among others, `berror` is used to display those messages shown at the end of the installation when the user must turn the key and press enter to reboot the system.

3.4.2 Operating System Levels in bi_main
Two important considerations throughout the `bi_main` script are the level of the operating system that is currently installed on the system (if any) and the level that is currenlty being installed. This information is used to determine the default and allowable installation methods in several places in the script.

The level of AIX that is currently installed on the system is determined in the shell script `Get_RVG_Disks` by using the command `blvset`. The `Get_RVG_Disks` script is described on page 100.

The level of AIX that is being installed is hard coded into the `bi_main` script, and is stored in the variable CURLEVEL. This variable is described at the start of `bi_main`.

It is important to note that the levels referred to in `bi_main` and returned by the `blvset` command may not neccesarily correspond exactly to the level of AIX being installed. In fact, the Installation process only allocates a new "level" to an AIX release or modification level if the user cannot upgrade from the previous release or modification level. For example, to move from AIX Version 3.2 to Version 4.1, it is neccesary to reinstall the system. It is not possible to upgrade simply by applying updates. However, when upgrading from AIX Version 4.1.2 to Version 4.1.4, the user can simply apply updates. The values of the CURLEVEL variable for each AIX release are shown in Table 8.

Table 8. CURLEVEL variable values	
AIX Release	**CURLEVEL value**
3.1 all update levels	3.1
3.2 all update levels	3.2
4.1.0	4.1.0
4.1.1 - 4.1.4	4.1.1

The most surprising aspect of this table is the distinction made between AIX Version 4.1.0 and the other releases of AIX Version 4. A reinstallation was required when upgrading from Version 4.1.0 since there were several changes made to kernel data structures, the standard C library and to the `installp` command. The updated libraries required the updated kernel to operate, which made an upgrade impossible without reinstalling.

3.4.3 Load Data Files

The first part of `bi_main` is shown in Figure 10 on page 99. The script starts by providing a debugging exit for the user by using the `restore` command to attempt to extract from diskette the file ./bi_main.debug and the shell script `./startup`. The file `bi_main.debug` must contain the string "bi_main.debug" at the start of the file. If both files exist, the script will execute the `startup` script before continuing. Don't forget to set the permissions of the startup file to make it executable. If only the `bi_main.debug` file exists, `bi_main` will `exec` a korn shell (`ksh`) to allow the user to perform manual actions. Since exec is used to call the `ksh` at this point, it is not possible to resume `bi_main` processing.

Note that at this point of the processing the console has not yet been configured. If you require the use of the screen, you must ensure that your startup script first calls `/usr/lib/methods/cfgcon`. This will prompt the user to select the console and will then configure the selected console. Alternatively, some basic feedback can be given to the user through the system LED (or LCD) display by using the `/usr/lib/methods/showled` command.

This exit is intended for use in the development and debugging of the installation process, however it could be used to debug or modify the installation process for your specific requirements—particularly the startup file option. However, as you will see, there are other possible modification points that may be more useful.
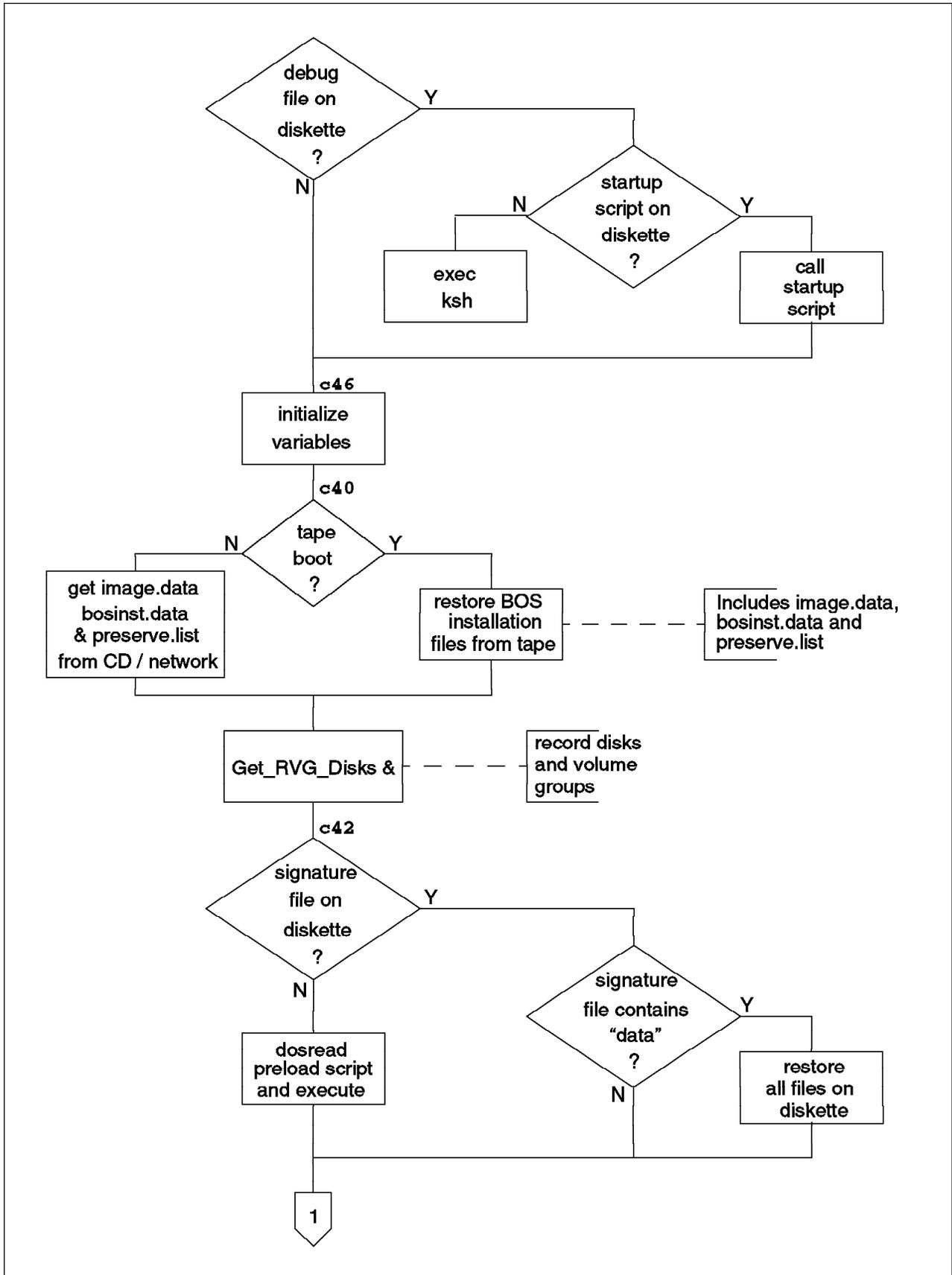
Figure 10. Loading Data Files - Part A.   Bold numbers show LED display values.

After initializing many variables and performing setup functions, in the case of a tape installation, `bi_main` restores the BOS installation files from the second tape image. In the case of CD-ROM or network installation, these files are already accessible from boot phase one. The script also copies the image.data, bosinst.data and preserve.list files from the boot device to the RAM file system. The contents of these files control the size of the file systems to be created during system installation, the use and default values of the installation menus, and the files to be preserved during the installation. Modifying the contents of these files provides a most flexible and powerful method for tailoring the installation process. These three files are described separately in 3.8.1, "bosinst.data file" on page 160 and 3.8.2, "image.data file" on page 163.

The `bi_main` script then makes a record of the disks in the system and the volume groups that reside on those disks. It does this by starting (in the background on systems with more than 8 MB of memory) the shell script `/usr/lpp/bosinst/Get_RVG_Disks`. This script builds a file /tmp/targetvgs. This file contains one line for each disk that is available on the system and lists:

- the Volume Group ID (VGID) for the volume group that includes the disk

- a flag with values:

    0   disk is not part of a volume group

    1   disk is part of the root volume group (rootvg)

    2   disk is part of another volume group

- the level of AIX installed if the disk is part of the rootvg

    See 3.4.2, "Operating System Levels in bi_main" on page 97 for information on the possible values of the level field. These values may not correspond exactly to the level of AIX installed on the system.

- the name of the disk (for example hdisk1)

- the location of the disk (for example 00-08-00-5,0)

- the size of the disk in megabytes

- a flag to show if the system is capable of booting from this disk

A sample targetvgs file is shown below:

```
0001271925aef8d4 1 3.2 hdisk0 00-08-00-0,0 382 y
00012719ccbb27dc 2 0.0 hdisk1 00-08-00-1,0 305 y
00009895939d17eb 2 0.0 hdisk2 00-06-00-00 817 y
00009895939d17eb 2 0.0 hdisk3 00-06-00-01 817 y
00009895939d17eb 2 0.0 hdisk4 00-06-00-02 817 y
00009895939d17eb 2 0.0 hdisk5 00-06-00-03 1022 y
```

The difference in format of the address fields in this sample is due to the fact that the first two disks are SCSI disks, while the remaining disks use the serial interface.

This file is then used later in the installation process for helping determine the target disks for the installation, and for determining the other volume groups that should be imported.

It is important to note that the targetvgs file is built after running `cfgmgr`. This means that the disks will have already had their names rediscovered in slot and address order. These are the names that the disks will have in the final migrated system. The phenomenon of changing device numbers and names is explained in 3.7, "Device Configuration" on page 156.

After starting `Get_RVG_Disks`, `bi_main` tries to restore a file `./signature` from diskette. If the file exists and contains the string "data" at the start of the file, all files on the diskette will be restored. This allows the user to provide their own image.data, bosinst.data or preserve.list files and is the most common method of tailoring the installation process (and probably the only officially sanctioned method of those described in this chapter). Since these files are read from diskette *after* restoring the files from the installation device, they will overwrite the original files allowing the user to install several systems from the same installation media, but using different image.data, bosinst.data and preserve.list files. If there is no signature file, `bi_main` will also try to `dosread` a file called `preload`. The `preload` file is assumed to contain `ksh` commands. If the file is found, it will be passed to a `ksh` to have its commands executed before proceeding.

The next part of `bi_main` is shown in Figure 11 on page 102.

If by this stage there is still no bosinst.data file, `bi_main` will now create a default file with minimal entries. It also checks to make sure that the image.data file exists before starting a daemon process `/usr/lpp/bosinst/datadaemon`. This daemon is used to read and update the bosinst.data and image.data files. It provides the services of parsing the stanza-style records in the files, and correctly updating the files when required. `bi_main` communicates with `datadaemon` by using the command `/usr/lpp/bosinst/bidata`. `bidata` provides a command line interface to `datadaemon` with options for reading the value of specific stanzas or entries in the files, and for updating specific entries.

Next `bi_main` sets the system console by first checking if a console is specified in the bosinst.data file. It also checks to determine if a non-prompted installation is specified before calling `/usr/lib/methods/cfgcon`. `cfgcon` will then check any specified console for validity and, if required, prompt the user to specify the console by pressing a number or function key. If non-prompted installation was selected, a screen containing three zeros ("000") will be displayed. This indicates to the user that they can still access the installation menus if desired by entering pressing the 0 key three times. There are also two other options. If the user enters three ones ("111"), `bi_main` will open a shell before the user menus. If the user types three threes ("333"), `bi_main` will ensure that turbo installation mode is not used. Turbo installation is a technique that is used in particular circumstances to provide faster BOS installation. For more information on turbo installation, see 3.4.10, "Turbo Installation" on page 123.

The script then starts an animated spinning line character to give the user visible feedback of action. It starts the program `/usr/lpp/bosinst/Animate` in the background to continuously rotate between the characters \, |, / and -. The process id of the `Animate` process is recorded so that it can be killed later.
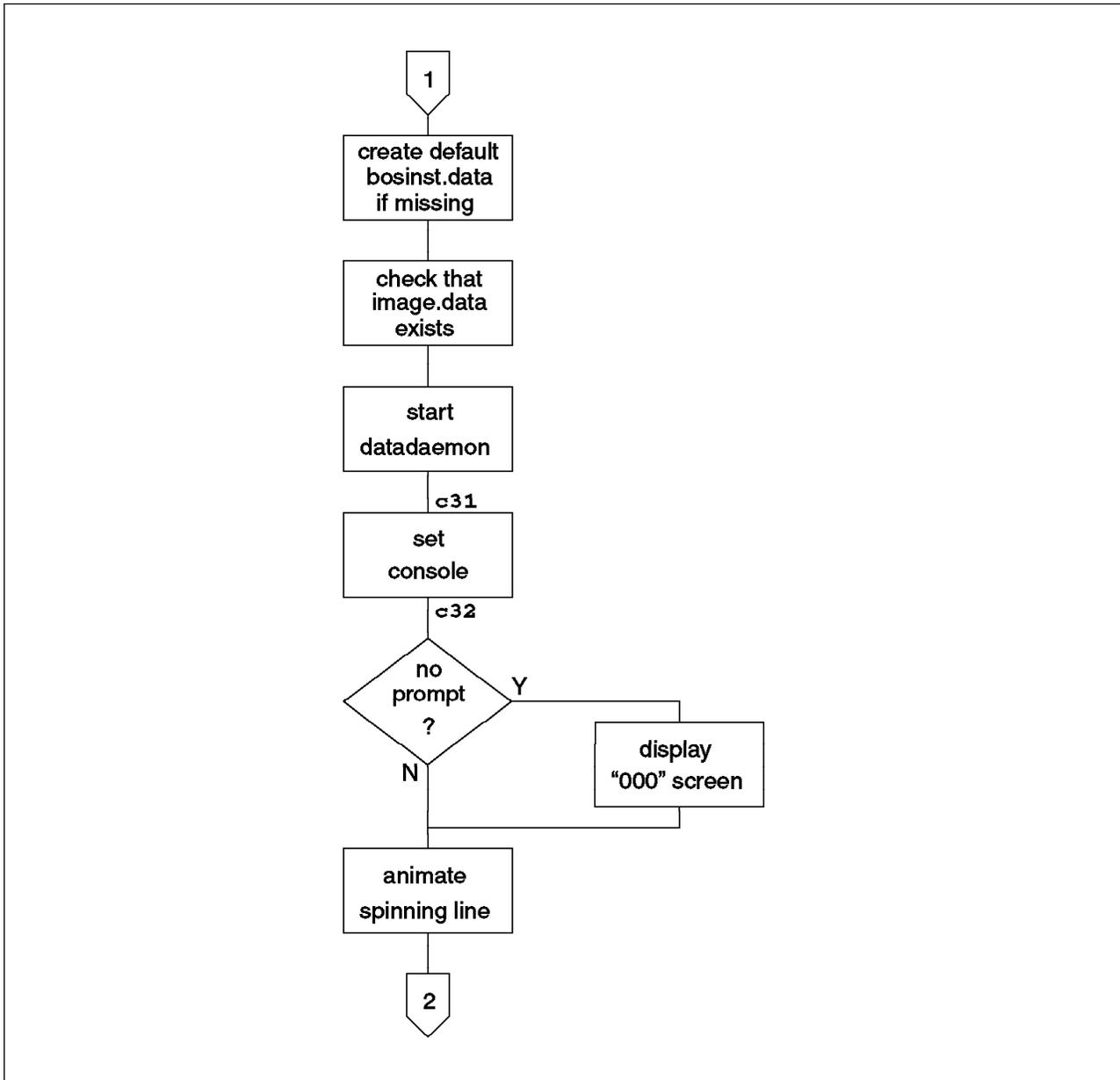
Figure 11. Loading Data Files - Part B. Bold numbers show LED display values.

3.4.4 Determine Target Disks and Installation Method

The next stage of `bi_main` is concerned with the selection and preparation of the disks that are to contain the rootvg of the newly installed or migrated system, and the selection of the installation method. This stage is shown in Figure 12 on page 104 and Figure 13 on page 105. Note that the two pages together form one double width flowchart. The program flow from Figure 11 enters the chart on the right hand page.

The logic for the selection of the target disks that will be used for the rootvg of the newly migrated system looks complex in the shell script, however in reality it implements just a few simple rules. The rules below are evaluated in order. The target disks will be determined by the first rule that matches.

1. If the target disk or disks were specified by the user in the bosinst.data file, they will be used as specified.

2. If there is an existing rootvg on any disk or disks in the system, they will be the default target disks.

3. If the system is a UniProcessor PowerPC system the target disks will be the disk with the highest SCSI address.

4. Otherwise, all available disks in the system will be selected as target disks.

> **Attention!**
>
> In the (admittedly rare) situation of a system with no root volume group but with one or more other volume groups, the disks that contain the other volume groups will be selected as target disks for installation. If an un-prompted installation is chosen in this situation, or if the user does not deselect these disks in the installation menus, it is possible to accidentally overwrite the other volume groups.

The logic for the selection of the installation method is easiest seen in Table 9.

Table 9. Installation Method Selection Logic		
Existing rootvg Operating System Level	**Method Specified in bosinst.data**	**Method to Use**
no existing rootvg	not specified	overwrite
	overwrite	as specified
	migration or preservation	overwrite display error message
V3.1	not specified	preservation
	migration	preservation display error message
	overwrite or preservation	as specified
V3.2	not specified	migration
	overwrite, migration or preservation	as specified
V4.1.0	not specified	migration
	overwrite, migration or preservation	as specified
V4.1.1 and above	not specified	preservation
	migration	preservation display error message
	overwrite or preservation	as specified

If the choices made are not valid, such as migrating an AIX Version 3.1 system or performing a preservation install on a system with no operating system installed, a more appropriate method will be selected and an error message will be set.
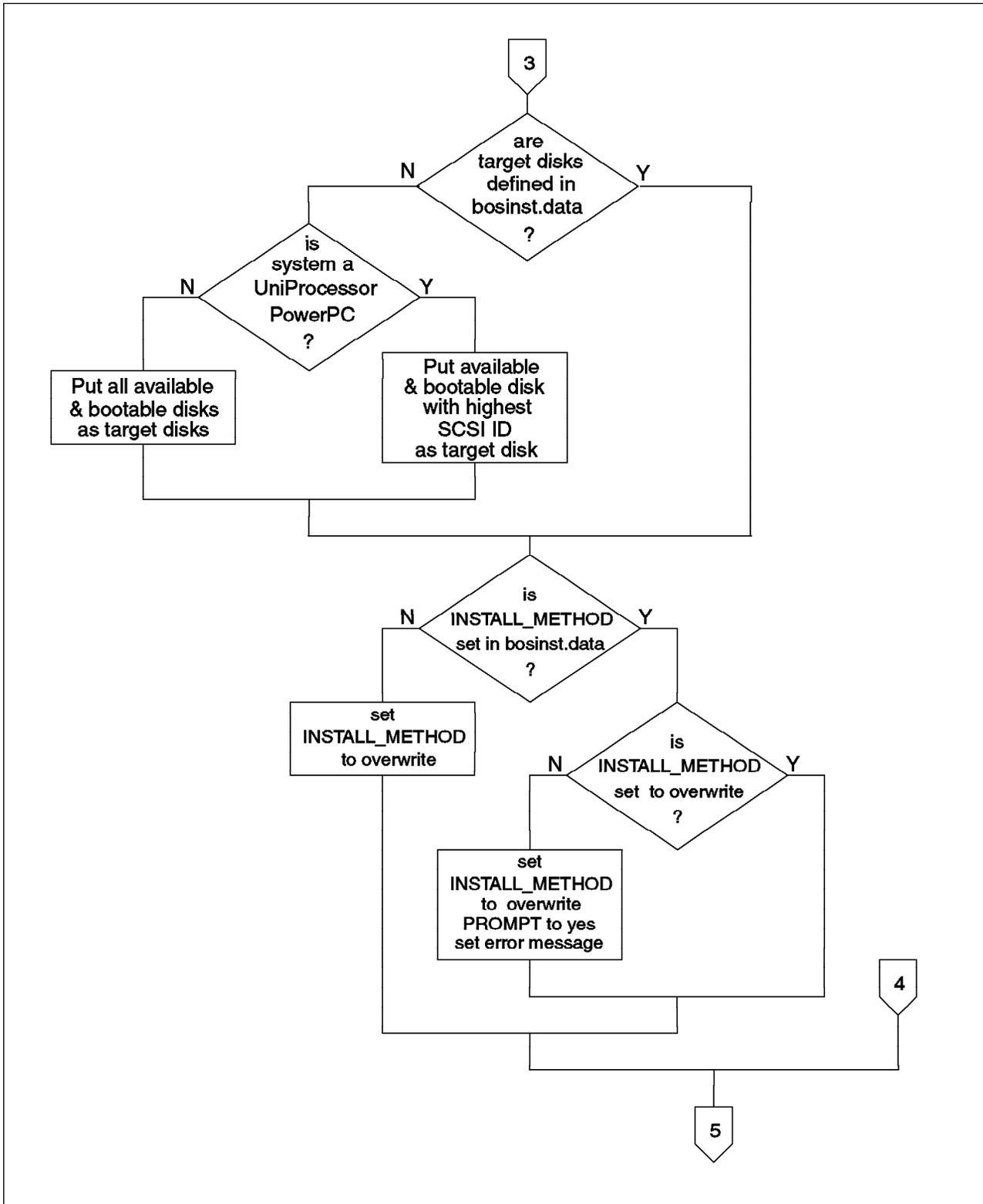
*Figure 12. Target Disks and Installation Methods - Left Side.*
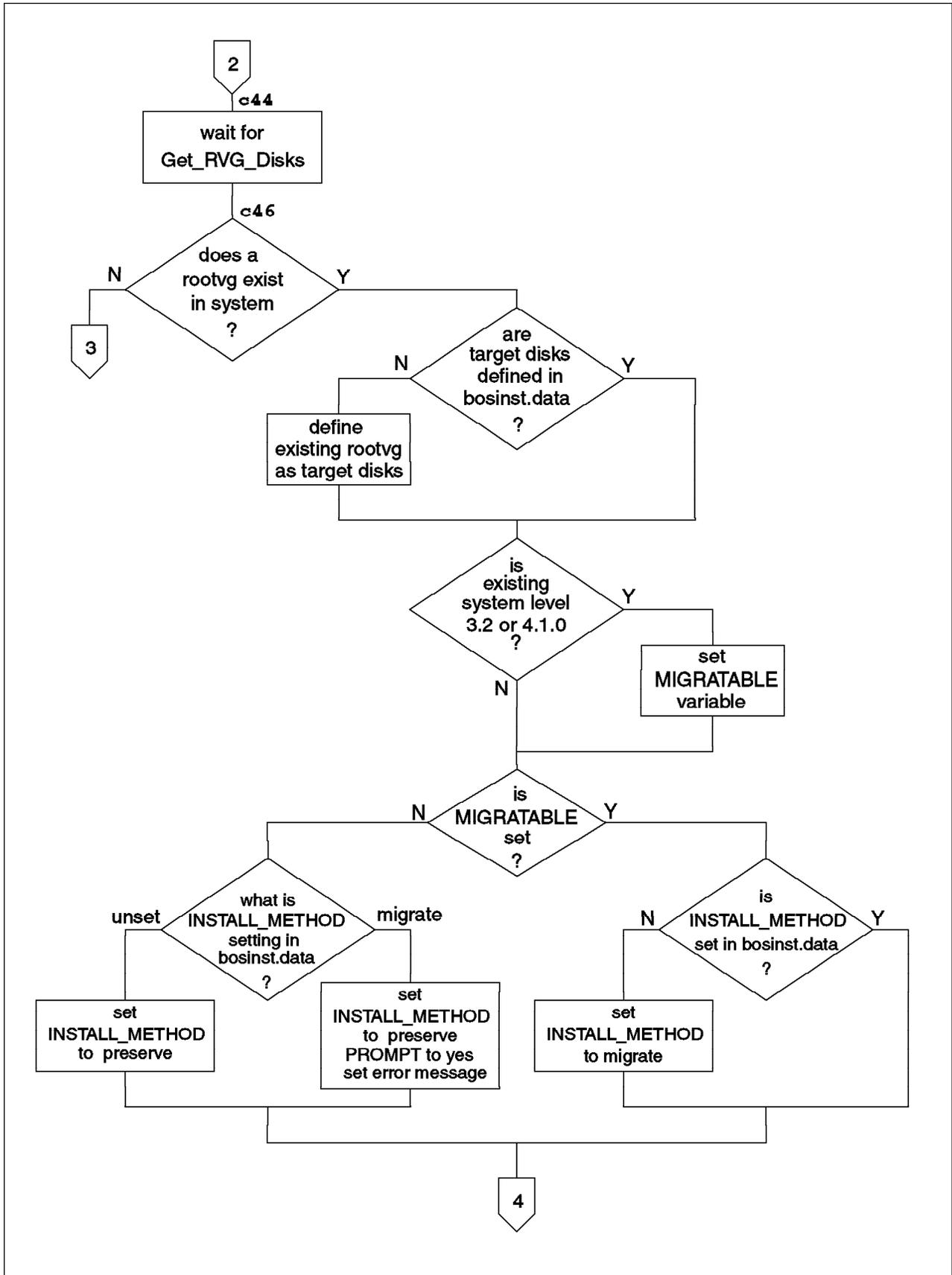
*Figure 13. Target Disks and installation Methods - Right Side.   Bold numbers show LED display values.*
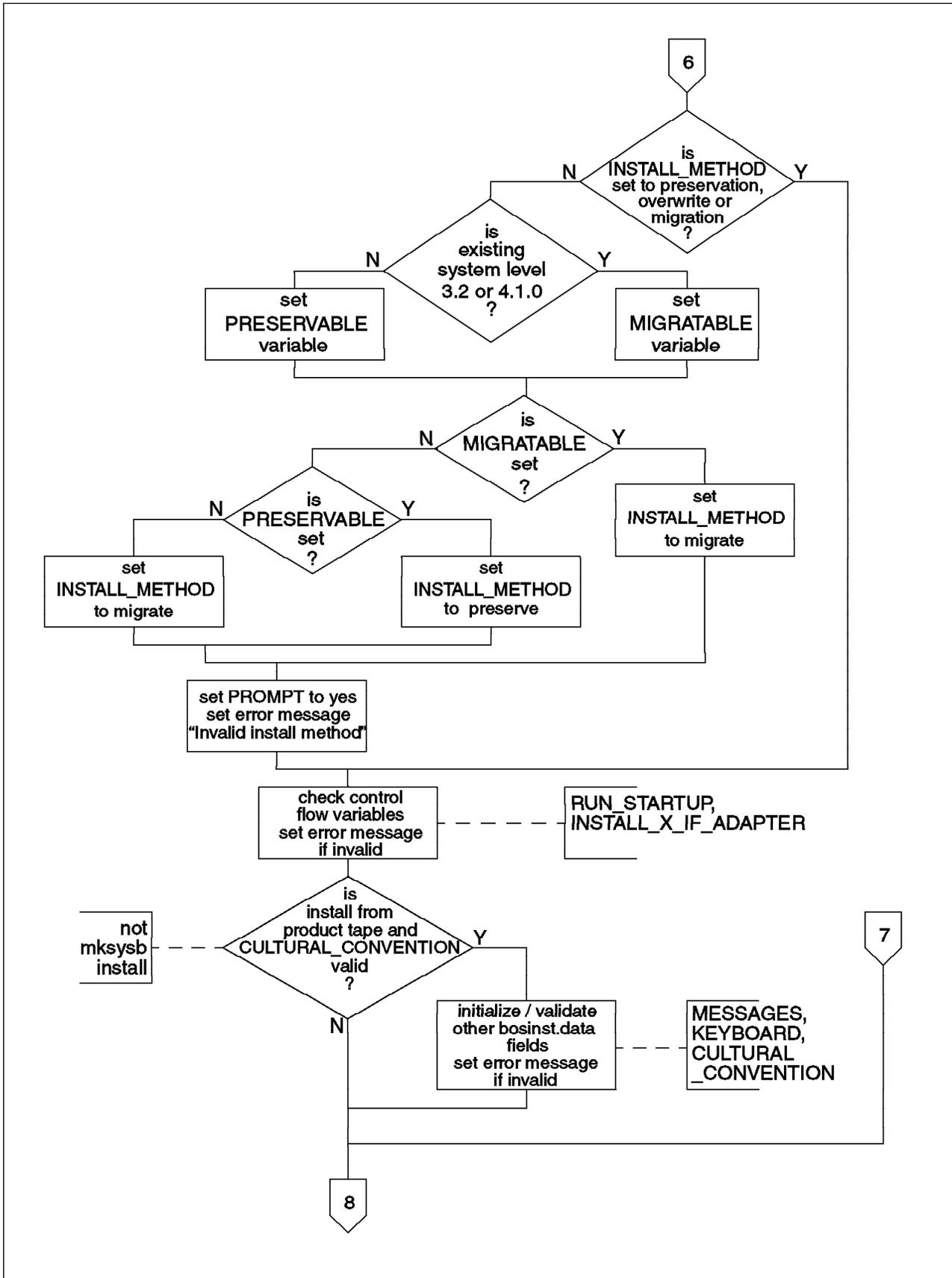
Figure 14. Validating bosinst.data - Part B.
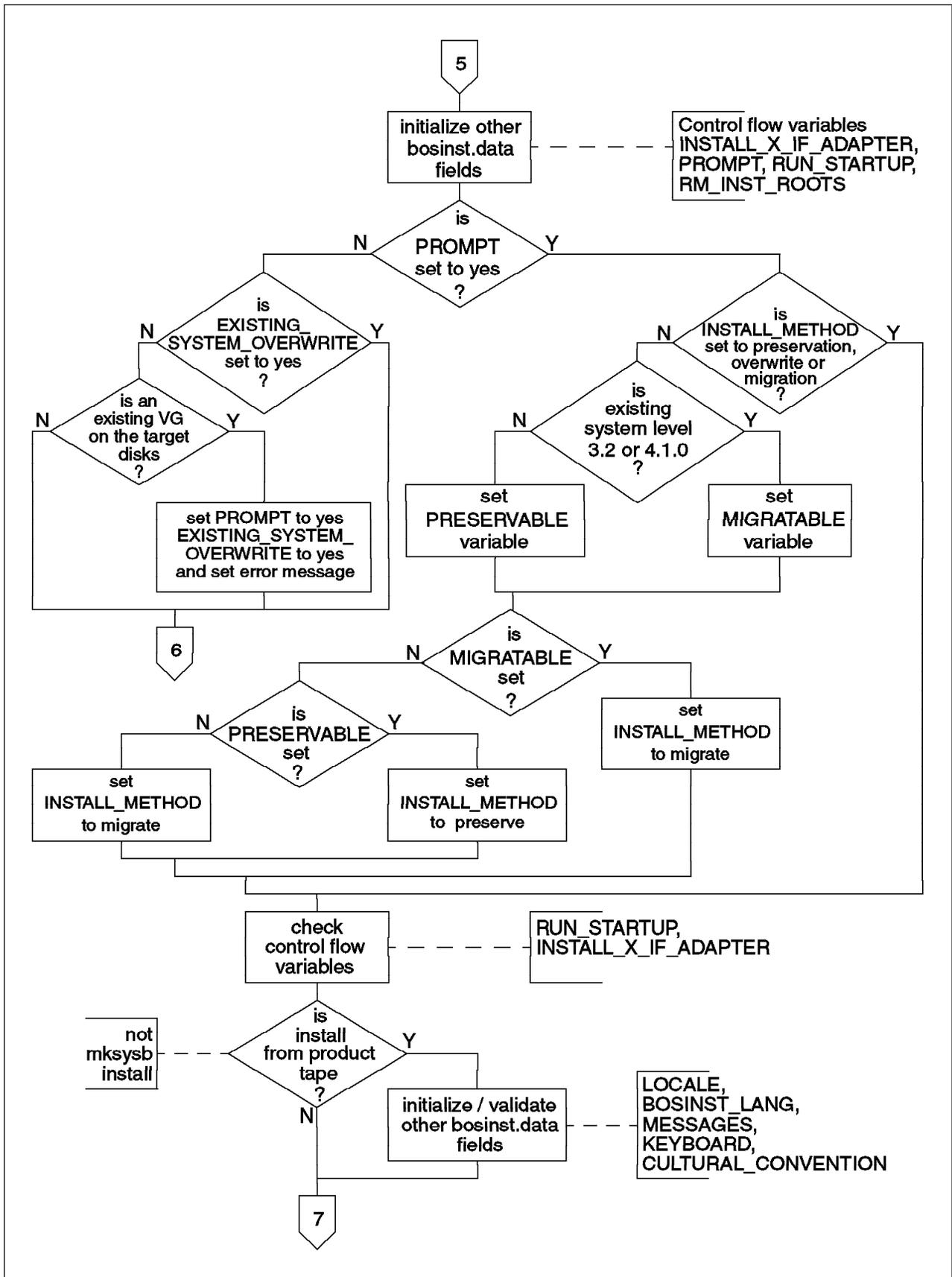
Figure 15. Validating bosinst.data - Part A.

3.4.5 Validate bosinst.data Values

The next stage of the `bi_main` script is concerned with validating the entries in the bosinst.data file, and is shown in Figure 14 on page 106 and Figure 15 on page 107. Similarly to the previous section, these two pages should be viewed together. The program flow enters on the right-hand page.

You will note that much of the logic in this section appears to duplicate the logic in the previous section for determining the migration method. This is true, however this section has one major difference. This section will protect against an invalid entry in the bosinst.data file.

You will also note that much of the logic in this section appears to be duplicated between 14 and 15. This is also true, however, again there is a major difference. These two flowcharts are mostly divided into two separate cases. Figure 14 illustrates the case of an un-prompted installation—that is, one where the PROMPT variable in bosinst.data is set to no. Figure 15 shows the case where the installation menus will be shown. Any errors that are found in the bosinst.data files are handled differently in these two cases.

In the case of a prompted installation, if an error is found in the bosinst.data file, the value is simply changed to the default setting for that field. When the user sees the value in the menus, they have the opportunity to ensure that it meets their requirements.

When the menus are not being used, it is necessary to inform the user of any errors in the bosinst.data file to ensure that there are no unexpected effects. Thus an error message will be set, and the PROMPT value will be set to yes to allow the user to see and act correct the error.

As well as validating the installation method, this part of `bi_main` also validates most other values in the Control_Flow stanza of the bostinst.data file. This includes the values for PROMPT, INSTALL_X_IF_ADAPTER, RUN_STARTUP, RM_INST_ROOTS and the international language support related entries. These and other bosinst.data variables are described in 3.8.1, "bosinst.data file" on page 160.

3.4.6 Validate image.data Values

The `bi_main` script then turns its attention to the image.data file, and to validating the entries contained in this file. Again, `datadaemon` and `bidata` are used to read and set the values of the fields in this file. This section of the processing is shown in Figure 16 on page 109
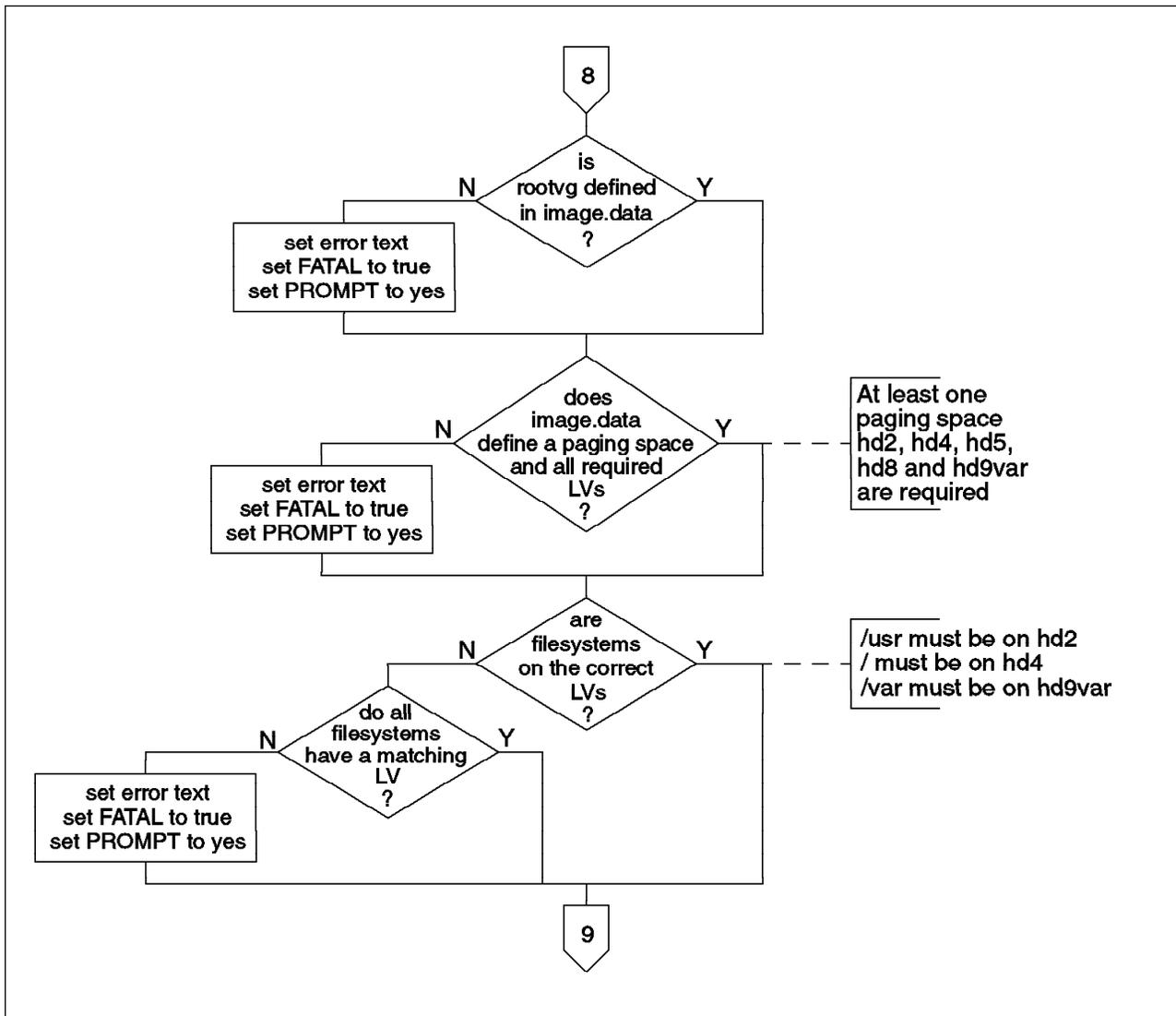
Figure 16. Validating image.data - Part A.

The rules implemented in this stage are fairly basic to the definition of the root volume group.

1. The image.data file must define a root volume group.
2. It must define at least one paging space.
3. It must define the Logical Volumes (LVs) hd2, hd4, hd5, hd8 and hd9var.
4. The basic system file systems must be defined on the correct file systems:
   - /usr must be defined on the hd2 logical volume.
   - / must be defined on the hd4 logical volume.
   - /var must be defined on the hd9var logical volume.
5. Every file system defined in the image.data file must have a corresponding LV defined.

Figure 17 on page 110 shows the next stage of the processing, which involves determining the target disks in the special case of installation from an image made using the `mksysb` command. In this case, the image.data file will contain a stanza source_disk_data which lists the disks that contained the rootvg in the system that was saved. This information is used as an additional decision criteria when determining the target disks for the installation.
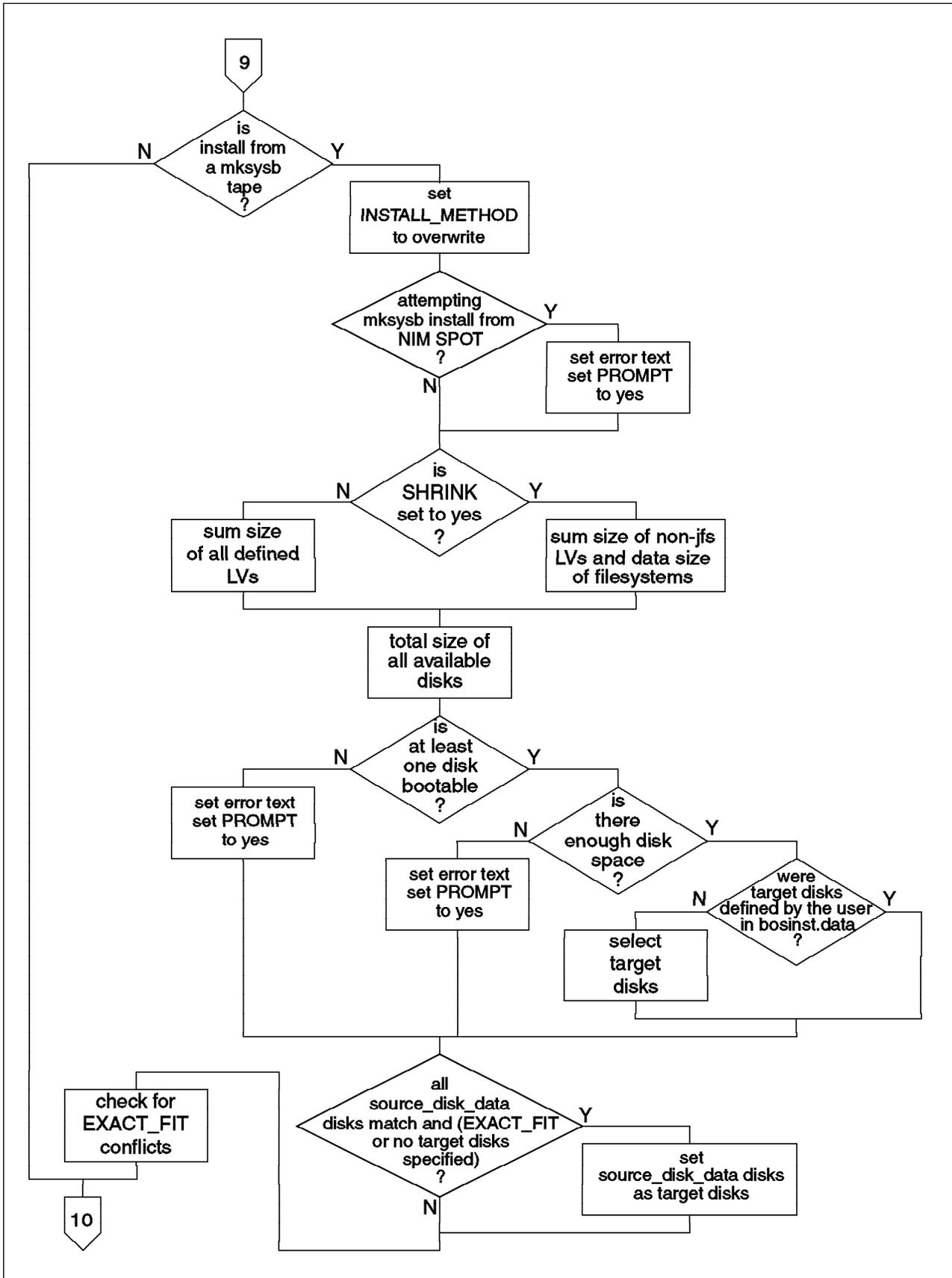
Figure 17. Validating image.data - Part B.

First, however, the system ensures that the INSTALL_METHOD variable in bosinst.data is set to overwrite and checks that the user is not attempting to perform a mksysb installation from a Network Installation Management (NIM) SPOT, as this is not possible. It then checks that the system contains enough disk in total to contain the complete system image. If the image.data variable SHRINK is set to yes, indicating that the user wishes all file systems to be created the minimum size required to contain the saved data, the script compares the total disk space available on the system against the total space used in the file systems and the space used by logical volumes that do not contain file systems (such as paging space and the boot logical volume). All of this information is contained in the image.data file in the mksysb image. If the SHRINK variable is set to no, or is unset, the script simply compares the total disk space against the total size of all defined logical volumes. If the system does not contain sufficient disk space to restore the mksysb image, an error message is set to be displayed in the installation menus.

If the target disks were defined by the user in the bosinst.data files, these selections will be accepted as specified. Otherwise, the system will select the target disks using the following criteria.

If the bosinst.data variable EXISTING_SYSTEM_OVERWRITE is set to no, indicating that any existing volume groups on the system should be preserved, the system will examine the disks in the system that have no associated Volume Group Identifier (VGID). It will ensure that at least one of these disks is bootable, and that the total disk space is sufficient to hold the mksysb image. Enough of these disks to contain the mksysb image will then be added to the target disks stanzas in the bosinst.data file.

If EXISTING_SYSTEM_OVERWRITE is set to yes, the script will first consider any disks that currently contain a root volume group. If these are not sufficient to contain the mksysb image, it will add any disks that are not part of a volume group. Finally, if required, it will add disks containing other volume groups until sufficient space is available. These disks will then be added to the target disk stanzas. The script will also ensure that the system is capable of booting from at least one of the selected disks.

As a last check, `bi_main` will examine the variable EXACT_FIT. This variable specifies that the disks listed in the source_disk_data stanza of the image.data file must match exactly the actual disks present on the system during installation. If the following conditions are met:

- The EXACT_FIT variable is set to yes

- The disks specified in the source_disk_data stanzas in the image.data file all exist in the system with the same size and location as specified

- The EXISTING_SYSTEM_OVERWRITE variable is set to yes, or the specified disks are not part of any volume group

then the source disks will be defined as the target disks. This is the only case when target disks specified in the bosinst.data file will be overridden.

Finally the system checks that the EXACT_FIT option was not specified illegally. If EXACT_FIT is specified, but the source disks do not exist on the system, or cannot be overwritten, or if both the EXACT_FIT and SHRINK options were set to yes, error messages will be set.

Having determined the target disks, the script must check that the stanzas defining these target disks in the image.data file are complete. It is possible that a user has specified an incomplete stanza or one listing a disk that does not exist. This stage is shown in Figure 18 on page 112.
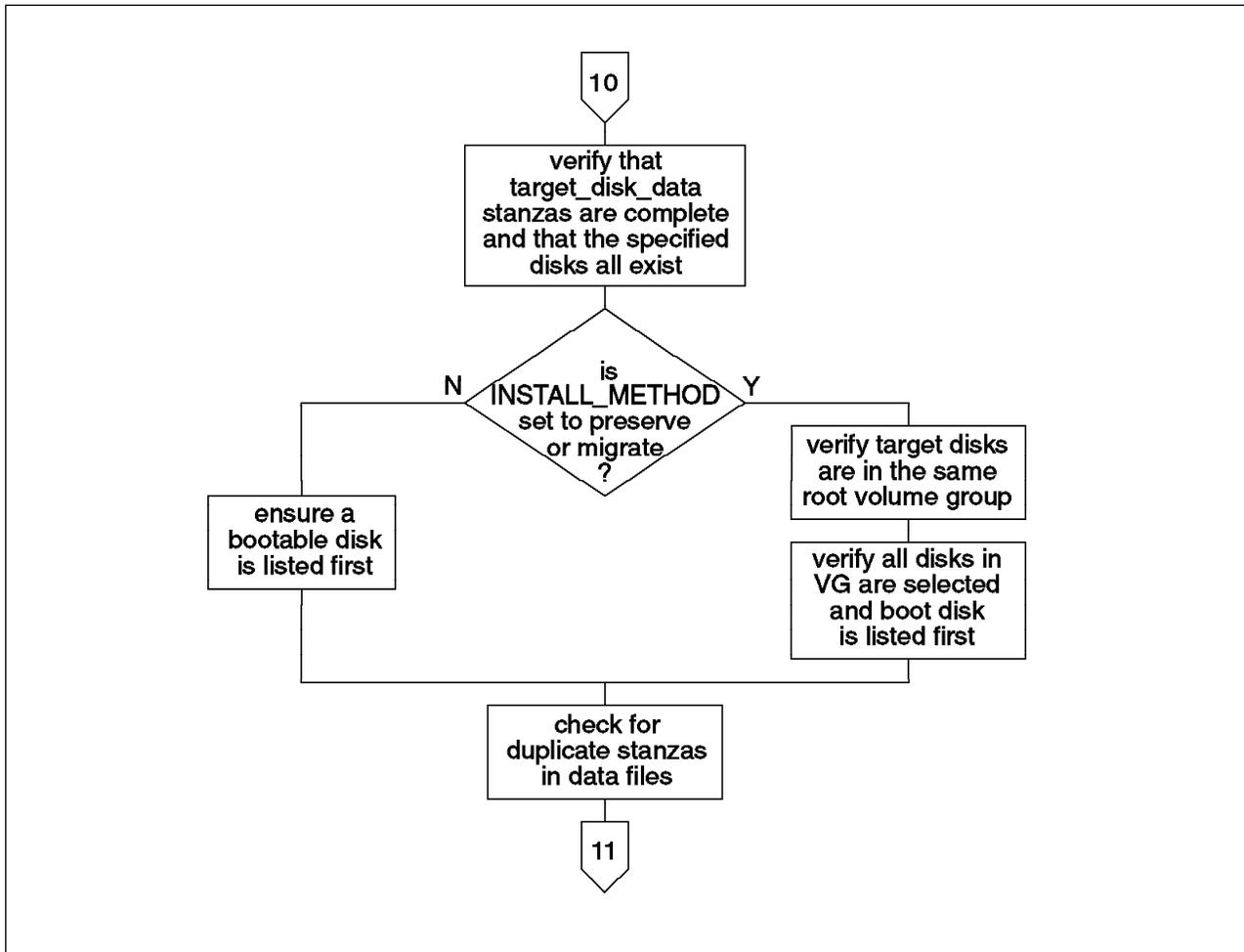
Figure 18. Validating image.data - Part C.

This section also checks that, in the case of an overwrite installation, at least one bootable disk is listed first in the target disks. This is the disk that will contain the boot logical volume hd5. In the case of a migration or preservation installation, the system will be installed onto an already existing root volume group. Thus the script verifies that all the listed target disks are in the same volume group, and that this group is a root volume group. It also ensures that all disks in the root volume group are selected, and that the boot disk is listed first.

3.4.7 User Menus

After processing the input data files and determining the target disks and installation method, the installation process has now reached the stage of displaying and processing the user menus. This stage of the process is shown in Figure 19 and Figure 20 on page 114.
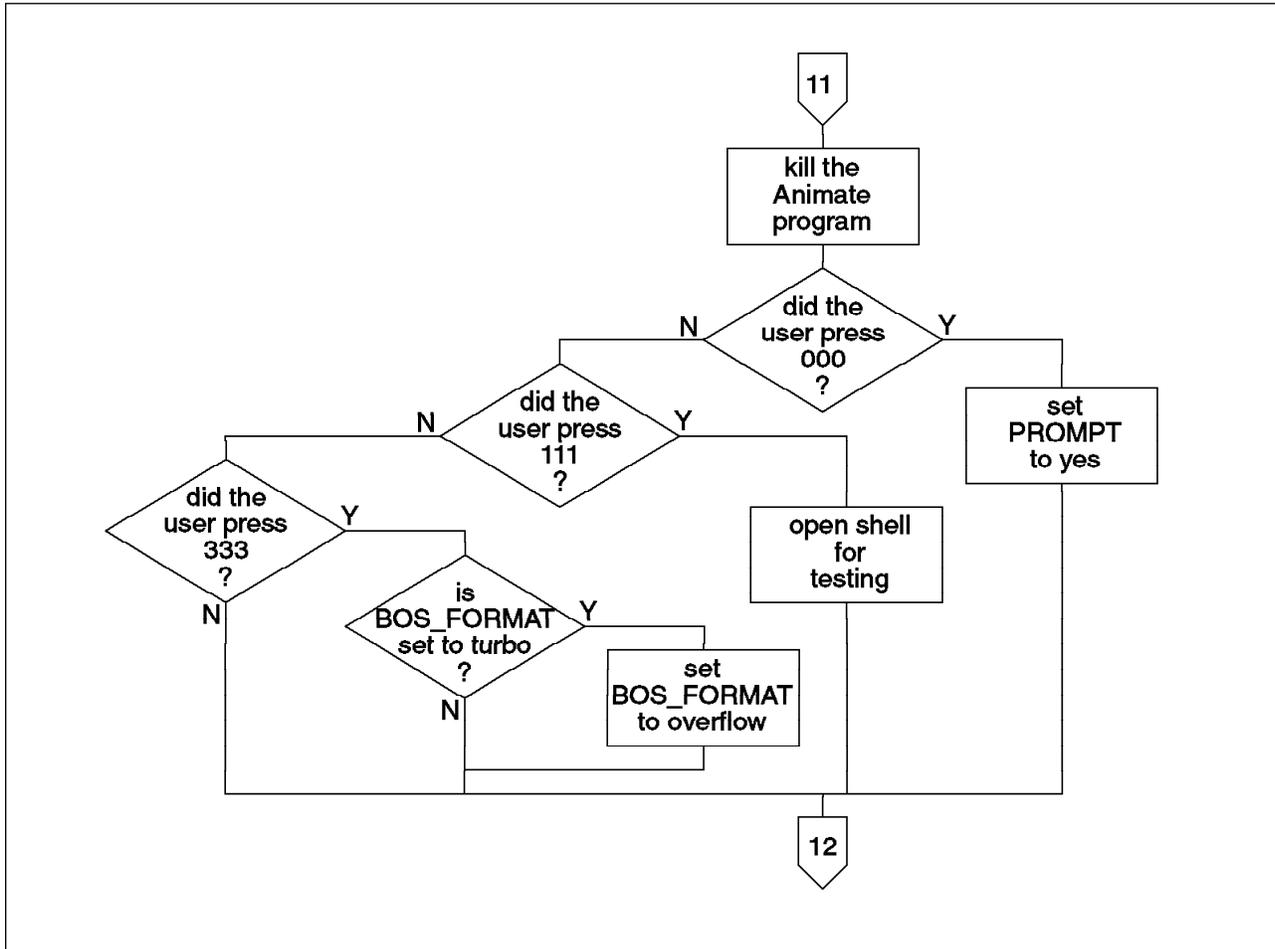
Figure 19. User Menus - Part A

The script first kills the `Animate` program that has been running to provide visible feedback of activity. In the case of a non-prompted installation, 000 was previously displayed to indicate that the user could press three zeros to change to set PROMPT to yes to use the menus. This test is made now. In addition, if the user had pressed 111, an interactive shell is opened for testing purposes. When the user exits the shell, processing will continue. If the user had pressed 333, the script will ensure that a turbo installation is not performed.

At this point the script starts a loop which it will leave only when all values have been entered or accepted through the menus and validated. In the case of an un-prompted installation, the script will skip directly to validating the values, however if any errors are encountered, PROMPT will be set to yes so that the menus will be used the next time through the loop.

Since the menu programs also read the data files, datadaemon is told to exit to ensure that any modifications are written to the files. The installation menus are then displayed, using defaults drawn from the information in the two data files. The
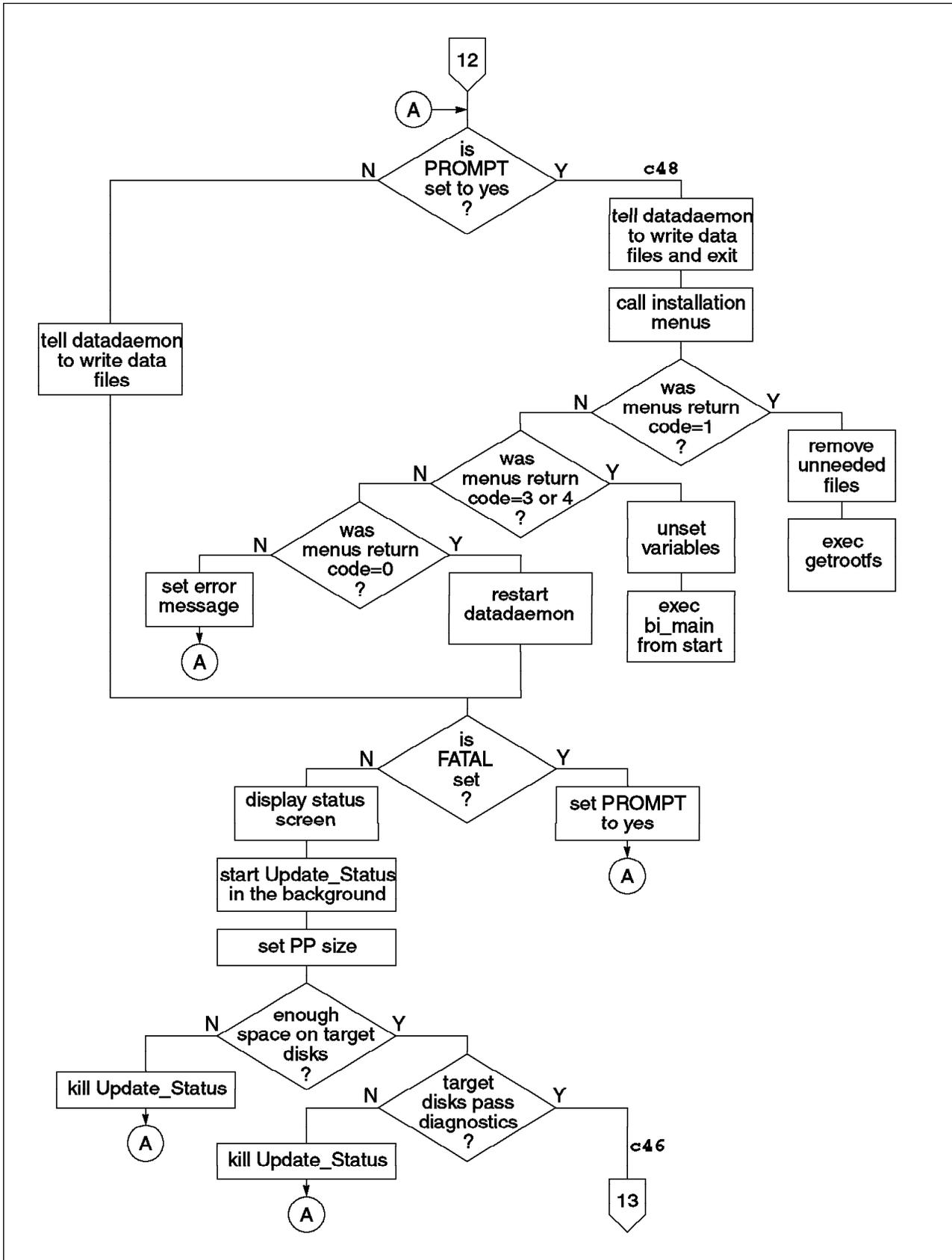
Figure 20. User Menus - Part B. Bold numbers show LED display values.

installations menus are shown in 1.4, "Migrating AIX" on page 33. Once the user completes the menus, the processing returns to the `bi_main` script which examines the return code of the menu program. Depending on the return code, the script will take the following actions:

0   The user has asked to proceed with installation.

The script will proceed to validate the results.

1   The user selected the option to start a maintenance shell.

Since the installation process is going to exit at this point, the script deletes some unneeded files. It then uses the `exec` command to call the `getrootfs` program. This imports and varies on the existing root volume group, and provides the user with a shell interface.

3 or 4   The user selected installation from a device that was not the boot device.

This is perfectly legal, however up until this time `bi_main` has been working on the assumption that the installation device was the same as the boot device. In order to process bosinst.data and image.data files from the installation media, `bi_main` uses the `exec` command to re-start processing from the beginning.

Other   An error occurred in the installation menus.

An error message will be set before returning to the menus.

To validate the results, the script checks that there is sufficient space on the selected disks to contain the installation image, and performs diagnostic tests on the disks. If all tests are satisfactory, a status screen is displayed, and the function `Update_Status` is started in the background to periodically update the status screen. Every 20 seconds `Update_Status` calculates the elapsed time, and reads a percentage complete figure from a file that is updated by the function `Change_Status`. `Change_Status` is called directly at various points throughout the installation with specific messages as parameters. `Update_Status` will display these messages in the correct position on the screen and update the progress indicators.

3.4.8 Prepare Target Disks

The first stage of disk preparation is common to all installation types and is shown in Figure 21 on page 116.

First a test is made to determine if a turbo installation is permitted. A turbo installation is *not permitted* if:

- If the installation method is migration

- If INSTALL_X_IF_ADAPTER is set to no

- If INSTALL_X_IF_ADAPTER is set to all, but console is not an lft

- If Trusted Computing Base (TCB) is to be installed

Here we have another opportunity to turn on a debugging flag that will cause the rest of the `bi_main` script to be traced into a log file. This trace can be particularly useful when tracking down installation problems. To turn on the trace flag, the following somewhat obscure procedure must be followed. This procedure is referred to by AIX developers as the "key dance".
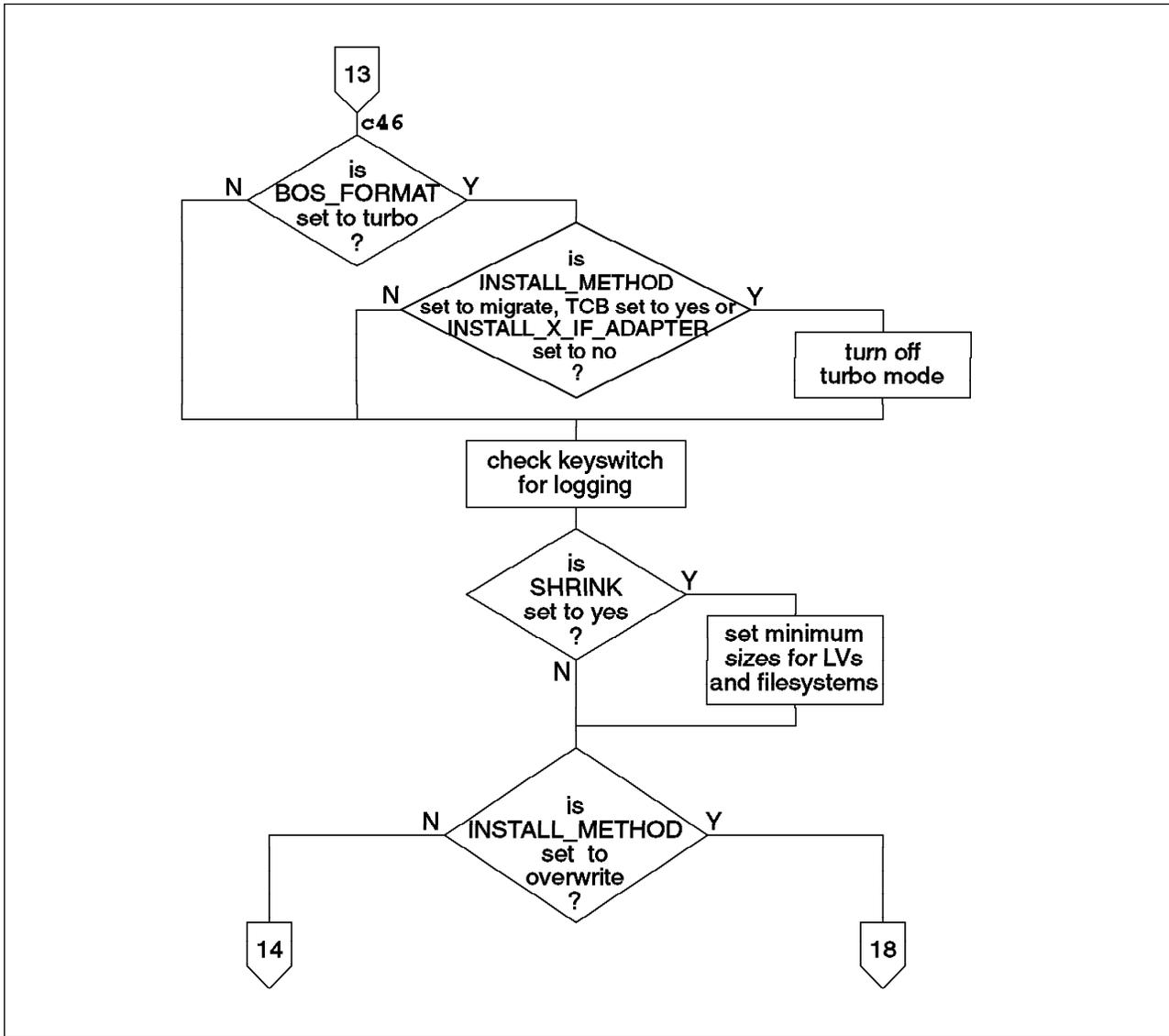
Figure 21. Preparing Target Disks.

1. Have the key in the secure position when the script reaches this point

2. The system LEDs will show the following characters for three seconds

3. During those three seconds turn the key to the service position

4. The system LEDs will show the following characters for three seconds

5. During those three seconds turn the key to the normal position

6. The system LEDs will be cleared and the tracing variable will be set.

This procedure cannot be used on systems without a key switch.

When the tracing flag is set, all other processing in the `bi_main` script will be traced using the ksh `set -x` setting, causing each command and its arguments to be displayed before it is executed. This output will be sent to the console to be viewed by the user.

If the tracing flag is not set, any output from the processing that is not specifically sent to the screen using the `Change_Status` routine will be sent to the log file /var/adm/ras/bi.log in the RAM file system. At the end of the migration this will be copied using the `alog` command to the /var/adm/ras/bosinstlog file on disk.

If the SHRINK variable is set to yes, the script will now set the size of the logical volumes and file systems to be created to the minimum size that will contain the data stored in those file systems.

From this point the processing branches, depending upon the chosen installation method.

3.4.8.1 Prepare Target Disks for Preservation or Migration
This section, shown in the three flowcharts Figure 22, Figure 23 on page 119 and Figure 24 on page 120, starts by accessing the existing root volume group and saving required information.
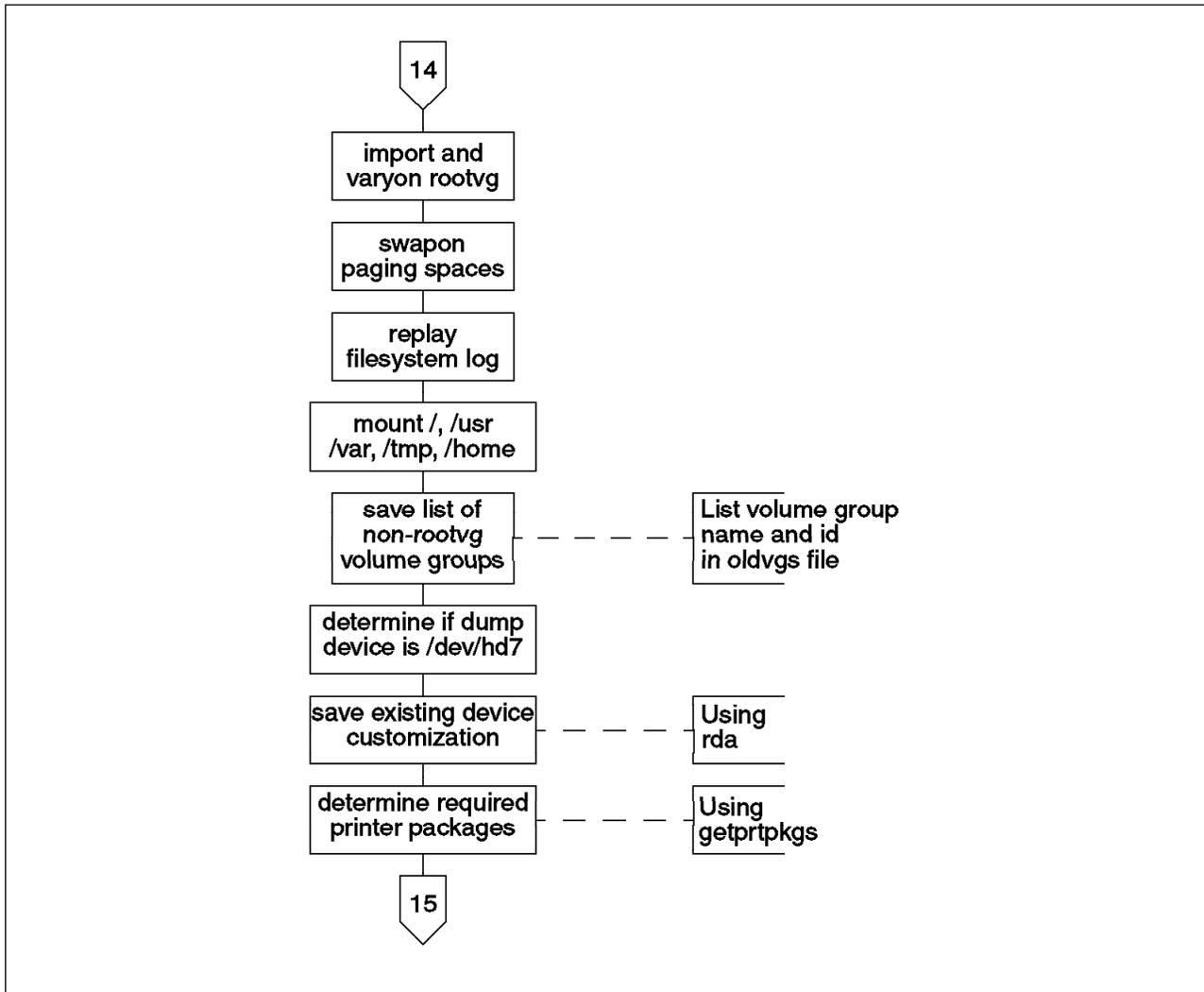
Figure 22. Preparing Disks for Preservation or Migration - Part A.

After importing and activating the rootvg, the system activates all rootvg paging spaces. This allows paging to occur and removes one of the great restrictions on the installation process of having to operate in just the system memory without paging. The existing disk file systems are also mounted under the /mnt directory.

The next important step is the building of a list of non-rootvg volume groups. The list is stored in the file /tmp/bos/oldvgs/.oldvgs and lists simply the volume group name and Volume Group ID (VGID). This information is obtained from the ODM Customized Attributes database, and will be used later in conjunction with the targetvgs file to import the non-rootvg file systems.

The script then saves the device information by using the `/usr/lpp/bosinst/rda` command. This extracts all of the relevant information from the ODM Customized Devices and Customized Attributes databases, storing it under the /tmp/bos/objrepos.inst directory. The shell script `/usr/lpp/bosinst/getprtpkgs` is
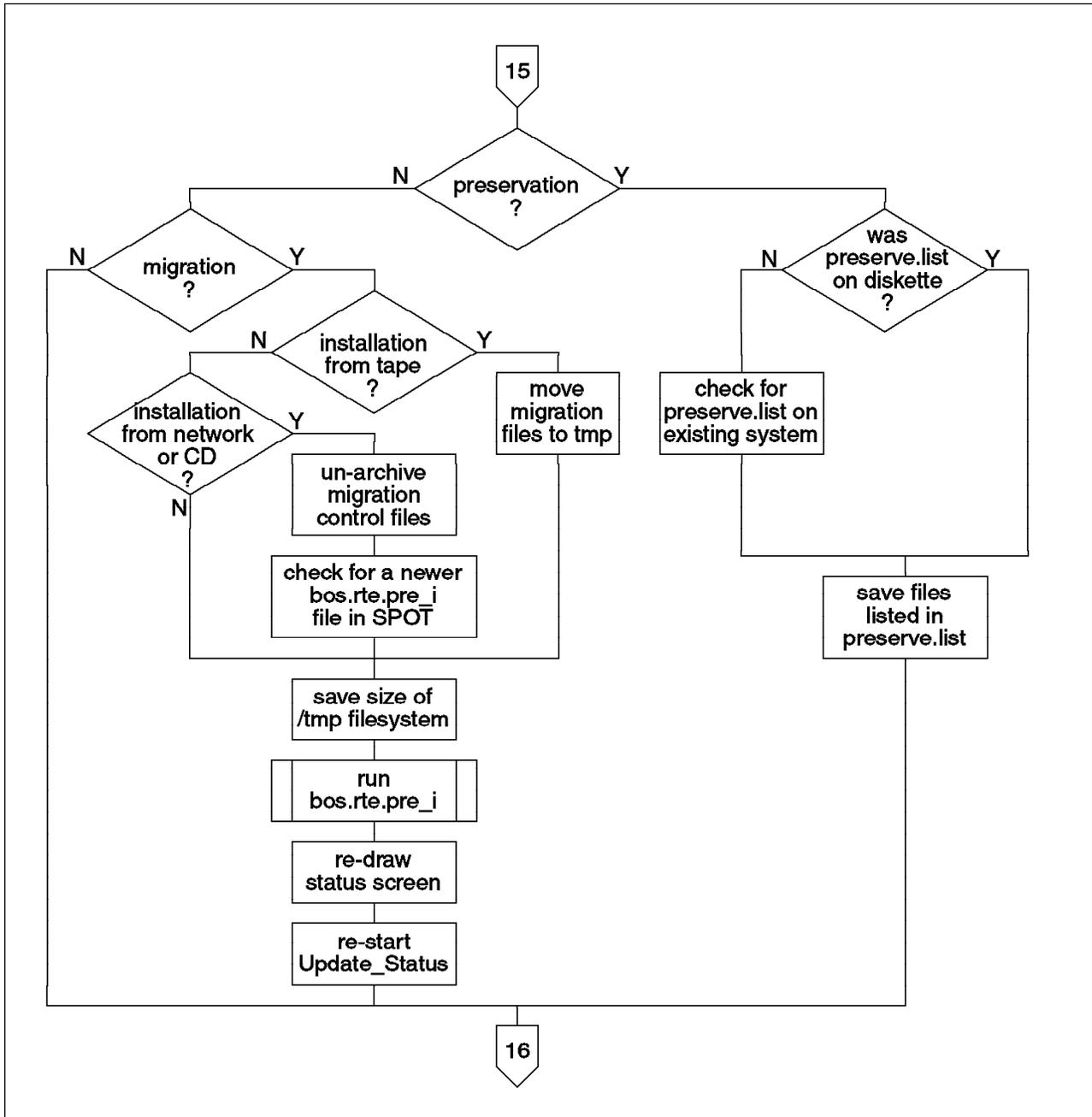
Figure 23. Preparing Disks for Preservation or Migration - Part B.

also called to build a list of the printer support packages that will need to be installed. We must know the required packages because AIX Version 4.1 no longer automatically installs support for every printer—support is installed selectively as required to save space.

Here the processing branches again. If a preservation installation is being performed, the script processes the preserve.list file and saves the listed files under the /tmp/bos directory structure. The script will also delete and recreate the hd2, hd4 and hd9var logical volumes and associated /usr, / and /var file systems.
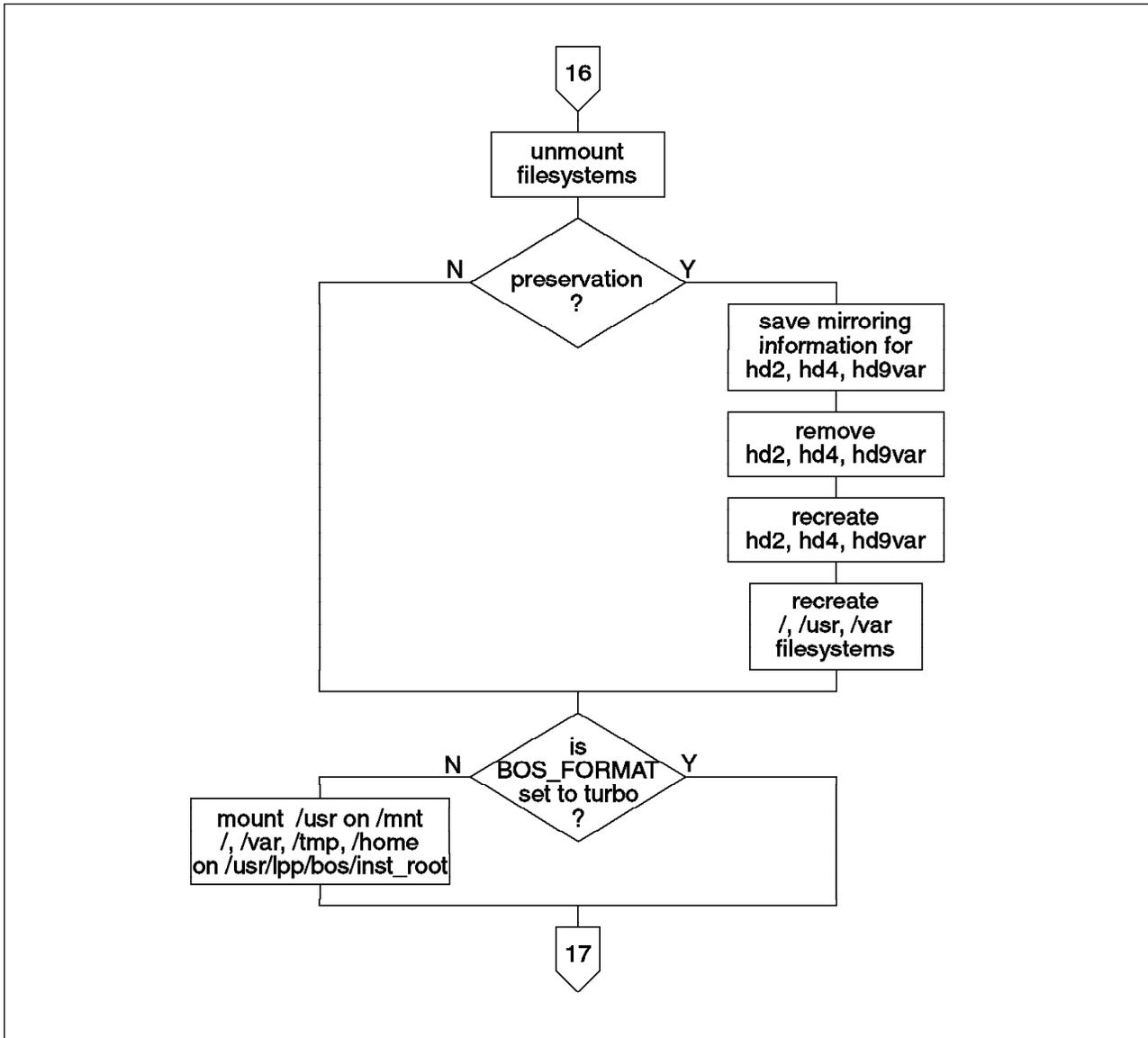
Figure 24. Preparing Disks for Preservation or Migration - Part C.

If a migration installation is in progress, the script retrieves the files:

- /usr/lpp/bos/bos.rte.cfgfiles
- /usr/lpp/bos/bos.rte.pre_i
- /usr/lpp/bos/bos.rte.*.rmlist
- /usr/lpp/bos/incompat.pkgs

and then runs the `bos.rte.pre_i` program. This program is described in 3.5.1, "Migration Pre-Installation Processing" on page 146.

Then, if the BOS_FORMAT variable is not set to indicate a turbo installation, the script will mount the disk file systems again. At this stage, the disk /usr file system is mounted on /mnt of the RAM file system. The /, /var, /tmp and /home file systems are mounted under the RAM file system's /usr/lpp/bos/inst_root directory. Any other file systems in the rootvg are then mounted under the /mnt tree if they are below /usr in the file tree, or under /usr/lpp/bos/inst_root otherwise.

3.4.8.2 Prepare Target Disks for Overwrite

As we can see in Figure 25, the processing required for a complete overwrite is much simpler. First the root volume group is created. Then the paging spaces, boot logical volume and other logical volumes and associated file systems are created. The paging spaces are activated once created to allow the system to start paging and thus remove the restriction on the virtual memory being limited to the amount of physical memory.
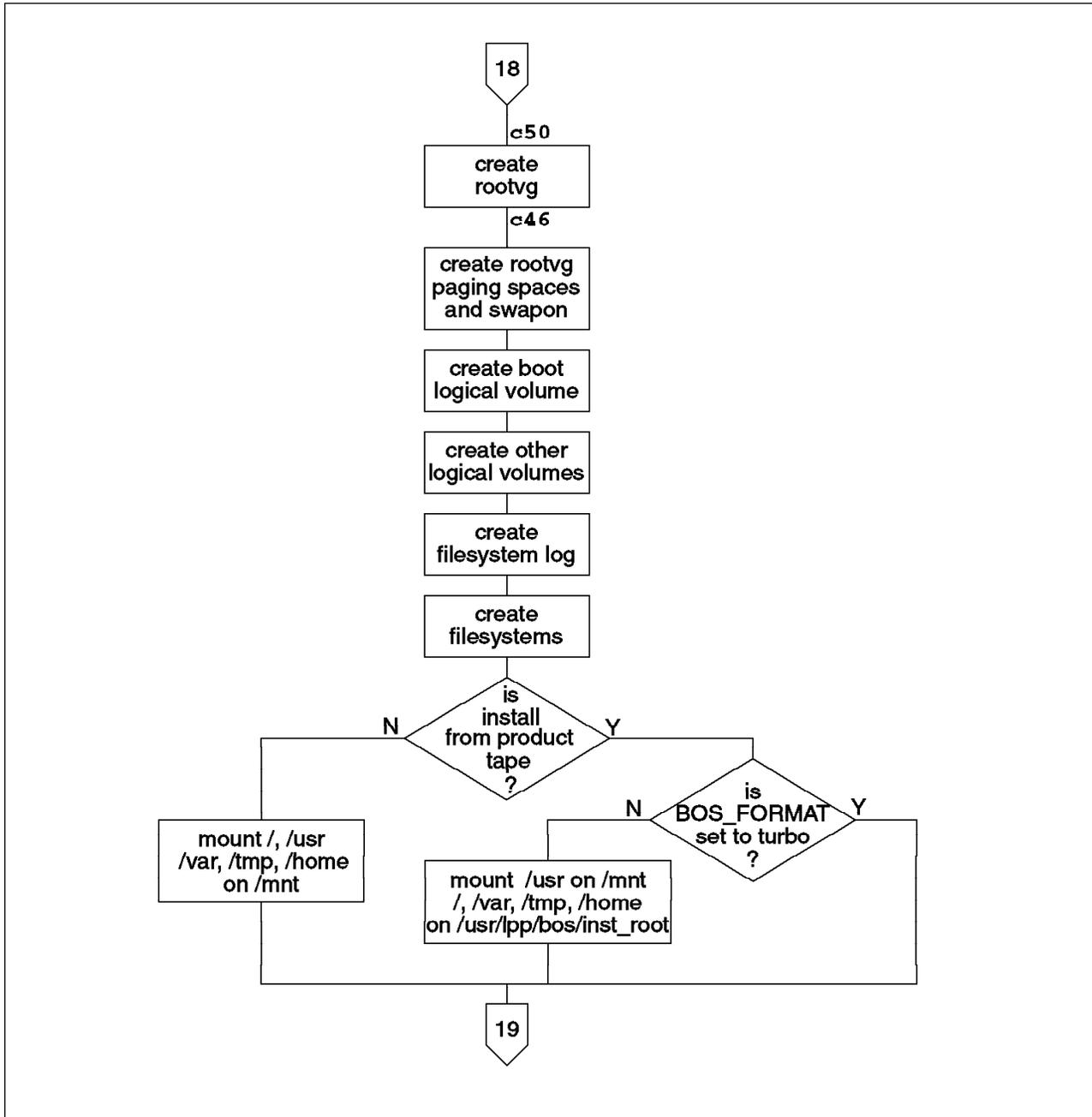
Figure 25. Preparing Disks for Overwrite. Bold numbers show LED display values.

If a turbo installation is not being performed, the newly created disk file systems will then be mounted on the RAM file system. Identically to the situation above, the disk /usr directory is mounted on the RAM directory /mnt. The disk file systems /, /var, /tmp and /home are mounted under /usr/lpp/bos/inst_root. Other file systems

are mounted as appropriate. The exception to this is the case of a mksysb installation, where all of the newly created disk file systems are mounted under the RAM /mnt directory.

3.4.9 Restore BOS Files

After a long preparation, `bi_main` has finally reached the stage of installing some code. This stage of the installation process is shown in Figure 26.
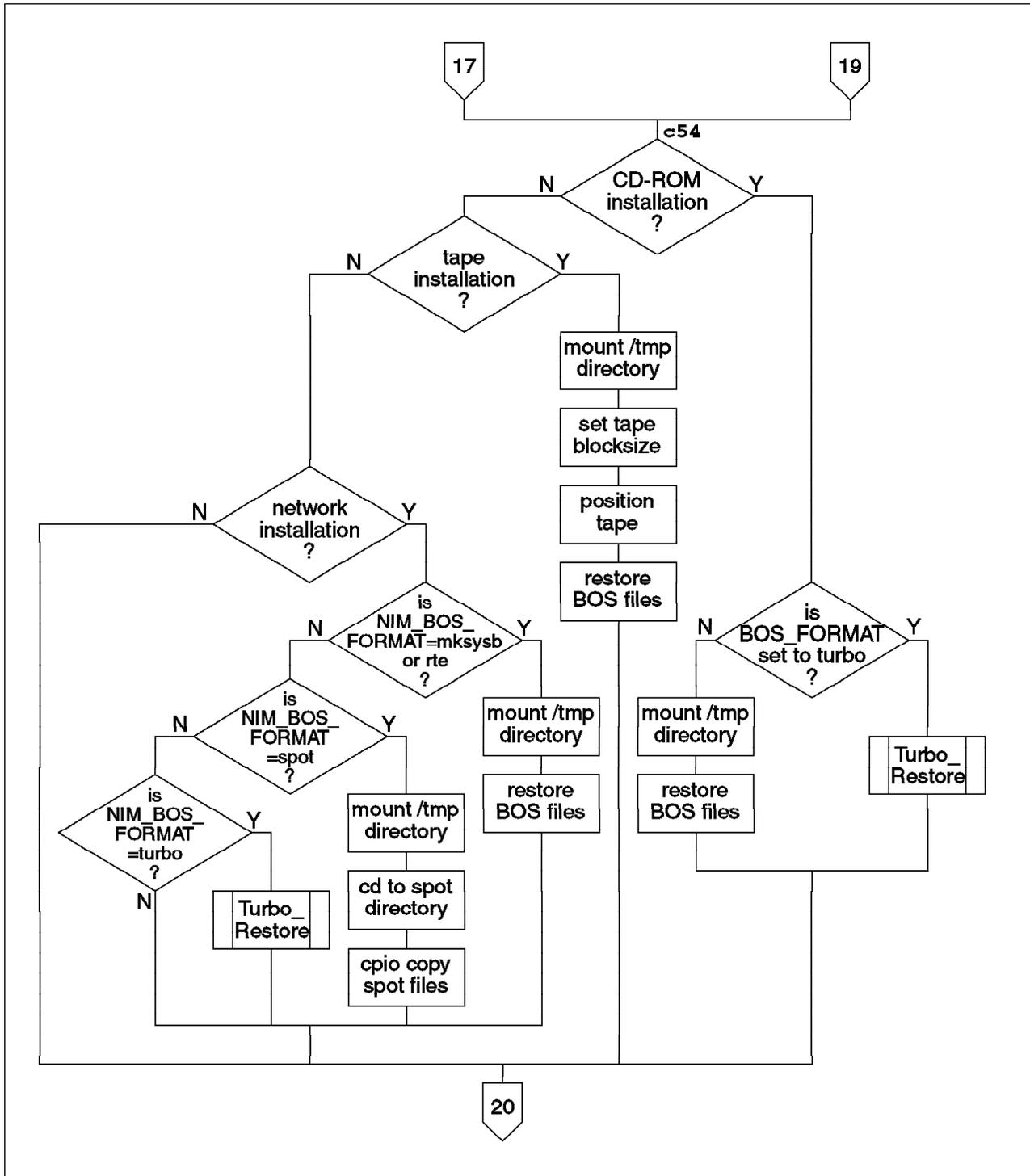
Figure 26. Restoring BOS Files. Bold numbers show LED display values.

First, the disk /tmp file system is mounted over the RAM /tmp directory to ensure that there is sufficient temporary space for the `restbyname` command. In most cases, the BOS files are then extracted from the backup format install image using the `restbyname` command. There are two exceptions from this rule. If a turbo installation is selected, then the turbo restore routine will be used. The turbo installation is explained in 3.4.10, "Turbo Installation" below. If the installation is from a NIM server and the NIM_BOS_FORMAT variable indicates an installation from a NIM SPOT, the script uses the `cpio` command to copy the SPOT files and directories to the system.

3.4.10 Turbo Installation

The turbo installation, shown in Figure 27 on page 124, is an optimized method of BOS installation that was introduced with AIX Version 4.1.3 which in some circumstances can greatly increase the speed of installation of the base operating system.

During the normal installation process, many separate filesets are installed. For each fileset, there are many actions that must be performed by the installation software including checking for prerequisites, checking for sufficient disk space and post-installation processing. Even with recent improvements to the installation software that has greatly improved the speed of these operations, this process still takes considerable time.

The turbo install operates by having on the installation media a complete, preinstalled and compressed hd2 logical volume. The installation is performed by simply creating an hd2 logical volume, then uncompressing the image over that logical volume. This volume includes all the files in the preinstalled /usr file system, as well as the files from the / and /var file systems. By using this preinstalled image, turbo installation avoids all of the time consuming checking and processing. In order to be completely hardware independent, the image contains no device driver filesets and does not include the bos.rte.up or bos.rte.mp packages. These packages will be installed as appropriate during post-installation processing.

The flowchart for a turbo installation is shown in Figure 27 on page 124. The uncompress of the logical volume is actually started in the background, allowing the script to perform other operations while it runs. `Update_Status` is called to provide feedback of the progress and copyrights and other user messages are displayed. Once the uncompress has completed, the file system mount point is set and `fsck` is run to check the file system is correct.

To complete the installation of BOS files in the other file systems, the /usr file system is mounted on the RAM /mnt directory, and the /, /var, /tmp and /home file systems are mounted on the RAM directory /tmpm, which is created for this purpose. The `cpio` command is then used to copy the files in these file systems directly from the uncompressed /usr file system where they are stored in the directory /usr/lpp/bos/inst_root. The files under this directory are then deleted. The space left free by these files in the /usr file system will later be reused as device filesets are installed.

Once the BOS installation is complete, the file systems will be unmounted, and remounted in the positions used in the normal installation methods. The disk /usr directory will be mounted on the RAM /mnt directory. The /, /var, /tmp and /home file systems will be mounted under the RAM /usr/lpp/bos/inst_root directory. Any other rootvg file systems are mounted under either position as appropriate.

```
          start uncompress
            of hd2 image

             update status

          display copyrights
            and messages

             update status

               wait for
              uncompress
             to complete

          copy /etc/filesystems
            from installation
                 media

            set filesystem
             mount point

               run fsck

           mount /usr on /mnt
          and /, /var, /tmp, /home
             on RAM /tmpm

           copy / and /var files
            from uncompressed
                 image

               unmount
              filesystems

           mount /usr on /mnt
           /, /var, /tmp, /home
          on /usr/lpp/bos/inst_root

                 return
```

Figure 27. Turbo Installation.

3.4.10.1 Limitations of Turbo Installation

Due to the nature of the turbo installation, it cannot be used in all situations. Since the turbo installation uses completely clean /, /usr and /var file systems, it cannot be used for migration installations.

In addition, it was necessary for the installation developers to choose the most common installation objects to include in the image. The image that was chosen includes AIXWindows support. Thus if a graphical installation is not selected, turbo mode cannot be used. In AIX Version 4 it is possible to remove LPPs after

installation. Thus it is possible to perform a turbo installation and then remove unneeded filesets however this may result in wasted space in the /usr file system.

The Trusted Computing Base (TCB) feature must be installed at the time of the initial BOS installation. Since TCB was not used when creating the hd2 image, turbo installation cannot be used when installing a system that is to use TCB.

Finally, turbo installation is used only when installing from CD. It is important to note here that although throughout the `bi_main` script you can see references to turbo installation over the network, this work was not completed. NIM does not yet support this feature and thus the variables will never be set to execute these sections of code.

3.4.11 Initialize Disk Environment

The operating system is now installed and you may feel that the job is done .... but wait—there's more!

Having installed the operating system, the script now starts preparing to transfer operations from the RAM file system that has served during the installation to the newly installed disk system. There are several steps that must be done in preparation for this change. These steps are shown in Figure 28 on page 126 and are mostly concerned with the device files in the /dev directory and the customized devices entries in the ODM.

In the case of a mksysb installation, the /dev directory on the disk is deleted and recreated to allow the devices that were discovered during the installation process to be cleanly moved to the disk /dev directory. However, if a device is the same both before the system backup and on the newly installed system, it is desirable to retain any customization that had been performed on that device. Thus the configuration of the devices is saved by using the `rda` command to the directory /tmp/bos/objrepos.inst on the disk. This will be processed later to merge the customizations. The dump device that was used on the backed up system is also saved at this time.

In the case of a migration from AIX Version 4.1.0 to AIX Version 4.1.4, the device definitions are compatible and thus can be preserved. To facilitate this, device files for those devices that existed before the migration and were discovered again afterwards will be copied from the disk /dev directory to the RAM /dev. The device entries in the disk /dev are then deleted in all cases before copying the entries from the RAM file system. The Customized ODM object classes (Cu*) are also copied.

Finally, the volume group information files stored under the directory /etc/vg are copied to the disk file system along with the console definition file /etc/consdef.

3.4.12 Change Mounts

This section, shown in Figure 29 on page 127, actually performs the remounting of the newly installed disk system over the top of the RAM file system.
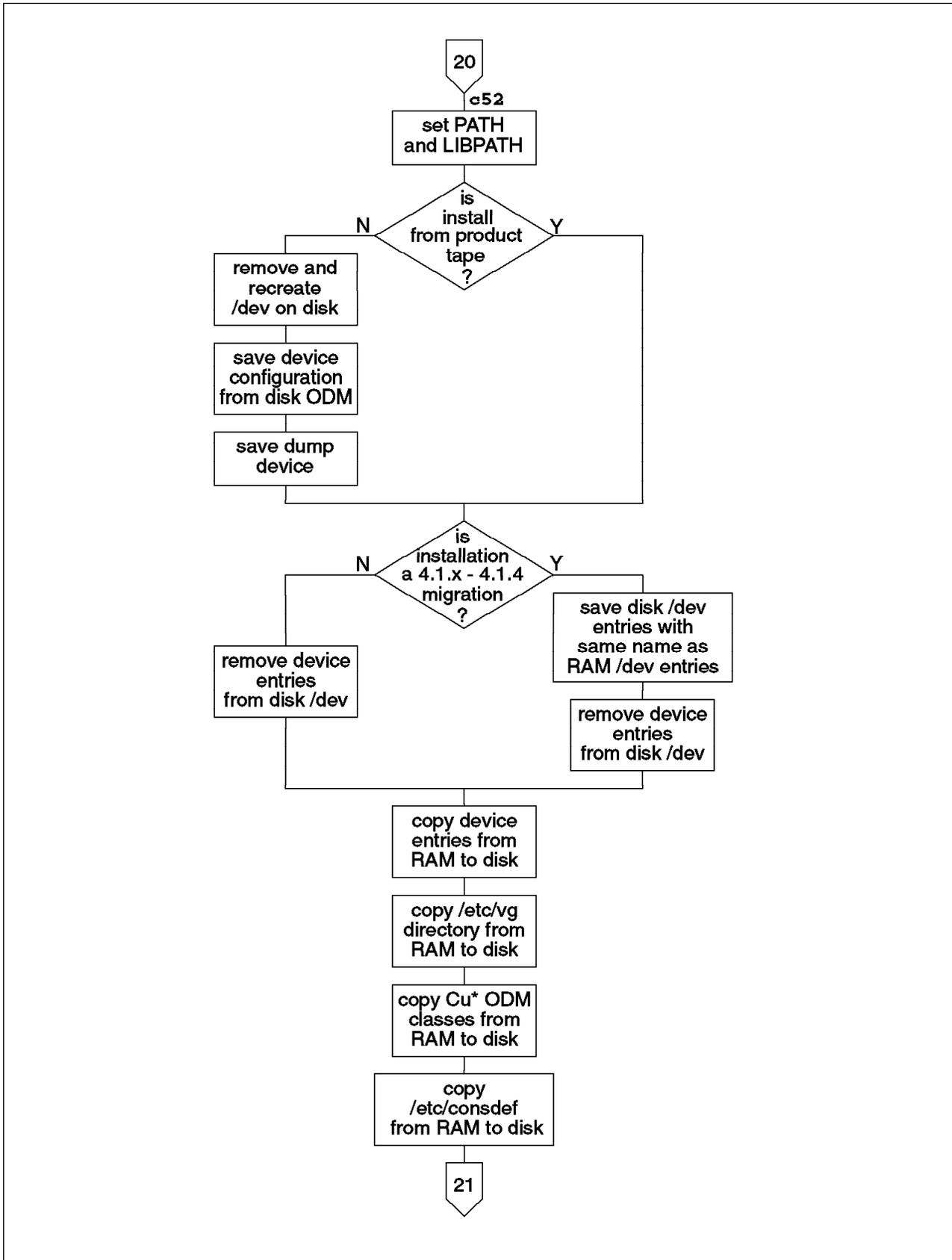
Figure 28. Initializing Disk Environment. Bold numbers show LED display values.
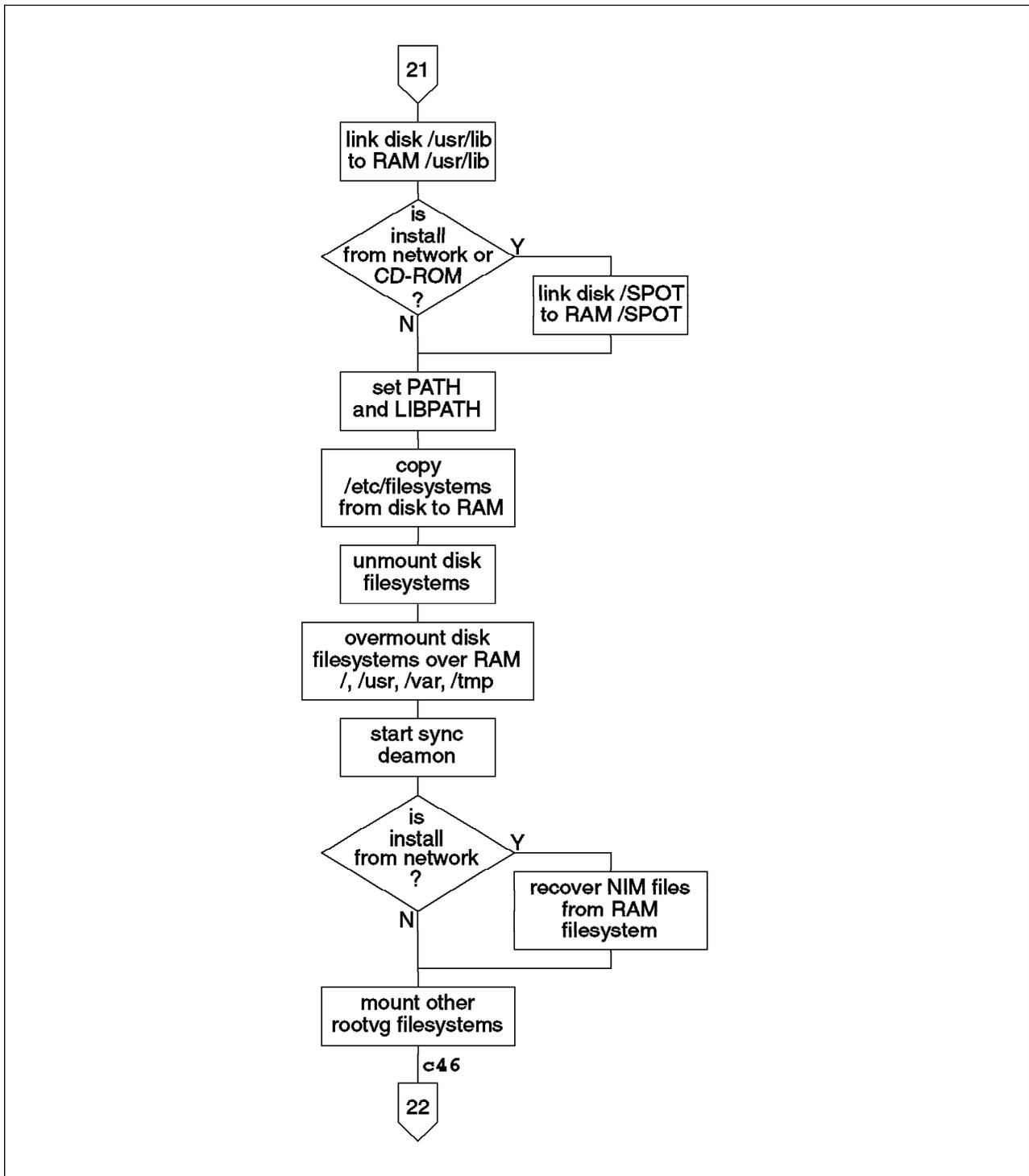
Figure 29. Changing Mounts. Bold numbers show LED display values.

Before starting, the script creates a link from the /usr/lib directory in the disk / file system (hd4) to point to /usr/lib in the RAM file system. This ensures that the mount command will be able to locate the required libraries at all times during the remounting process. Similarly, in the case of network or CD-ROM installation, the /SPOT directory on disk is also linked to the RAM /SPOT directory. The PATH and LIBPATH are both reset to ensure that only the commands from the RAM system are run.

To allow the appropriate file systems to be identified, the /etc/filesystems file is copied from the disk to the RAM file system. The disk file systems are then all unmounted. The system file systems /, /usr, /var and /tmp are then remounted in their usual positions—overmounting the corresponding RAM file systems. Since we will now be running from the disk file systems, the sync daemon is started to ensure that they are kept correctly updated.

For network installations, `bi_main` now calls the program named in the variable NIM_BOSINST_RECOVER. By default, this is the program `/usr/lpp/bos.sysmgt/nim/methods/c_bosinst_env`. Up until this point of the installation the various resources needed by NIM have been present in the RAM file system. As NIM is soon to lose access to the RAM file system, this routine sets up the required files in the disk file systems.

The last stage of changing the mounts consists of mounting of other file systems in the newly installed root volume group.

3.4.13 Post-Installation Processing

The `Post_Install` function is long (around 900 lines) and varied, so before starting this chapter we recommend your find a comfortable chair and a hot cup of coffee.

There are many steps that must be taken after the BOS files have been installed to complete the process of creating a system. These overall steps are shown in Figure 30 on page 129 through Figure 39 on page 140, though for clarity we have broken the function into four parts.

We start the post-installation process by setting the installation time in the ODM of the bos.rte filesets that were just installed.

If a migration or preservation installation is being performed, we then check if /dev/hd7 is being used as the dump device. If this device is being used it will be removed. Later in the post-installation process the dump device will be set to point to the paging space.

At this point, in a migration installation, the migration specific post-installation processing script `bos.rte.post_i` is extracted from the liblpp.a object file and executed. This script is detailed separately in 3.5.2, "Migration Post-Installation Processing" on page 149. Meanwhile, in a preservation installation, the files listed in preserve.list that were earlier saved to /tmp/bos are now restored to their correct positions on the disk file systems. If preserved system was at AIX Version 3.1, the /etc/filesystems file must be modified before being replaced. All file systems except / and /usr will be retained. The /, /usr and /var file system definitions will be copied from the /etc/filesystems file created during installation. In the case of a migration installation, the bos.rte.up or bos.rte.mp fileset will be deleted so that the appropriate fileset can be installed later.

At this stage, the installation branches, depending if the installation is from a product tape or a mksysb system backup. A product tape is an original software distribution tape. A product tape installation can be either an overwrite, preservation or migration installation.
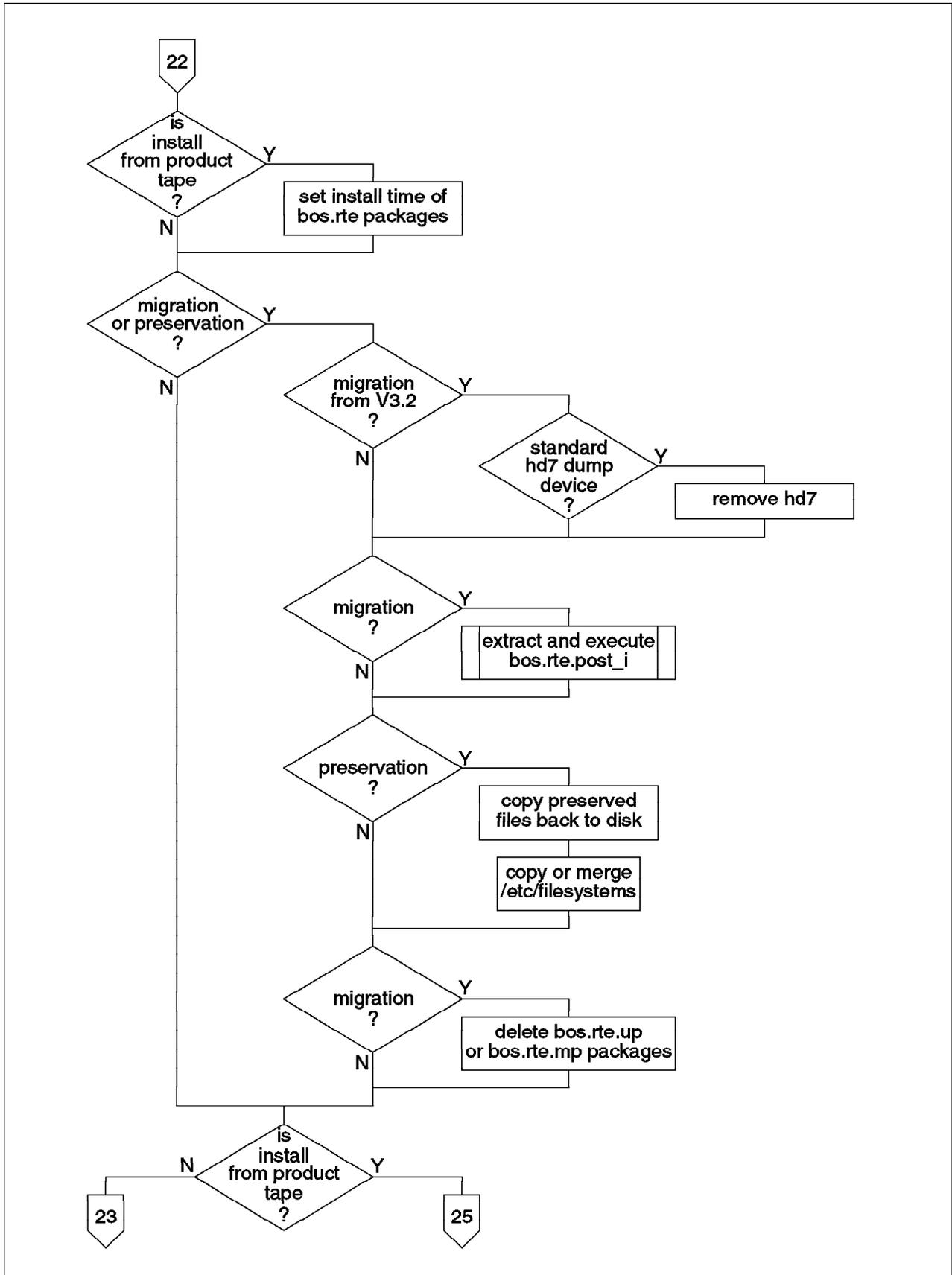
Figure 30. Post-Installation Processing.

3.4.13.1 Post-Installation Processing for mksysb Installations

This is by far the simplest of the two options and is shown in Figure 31.
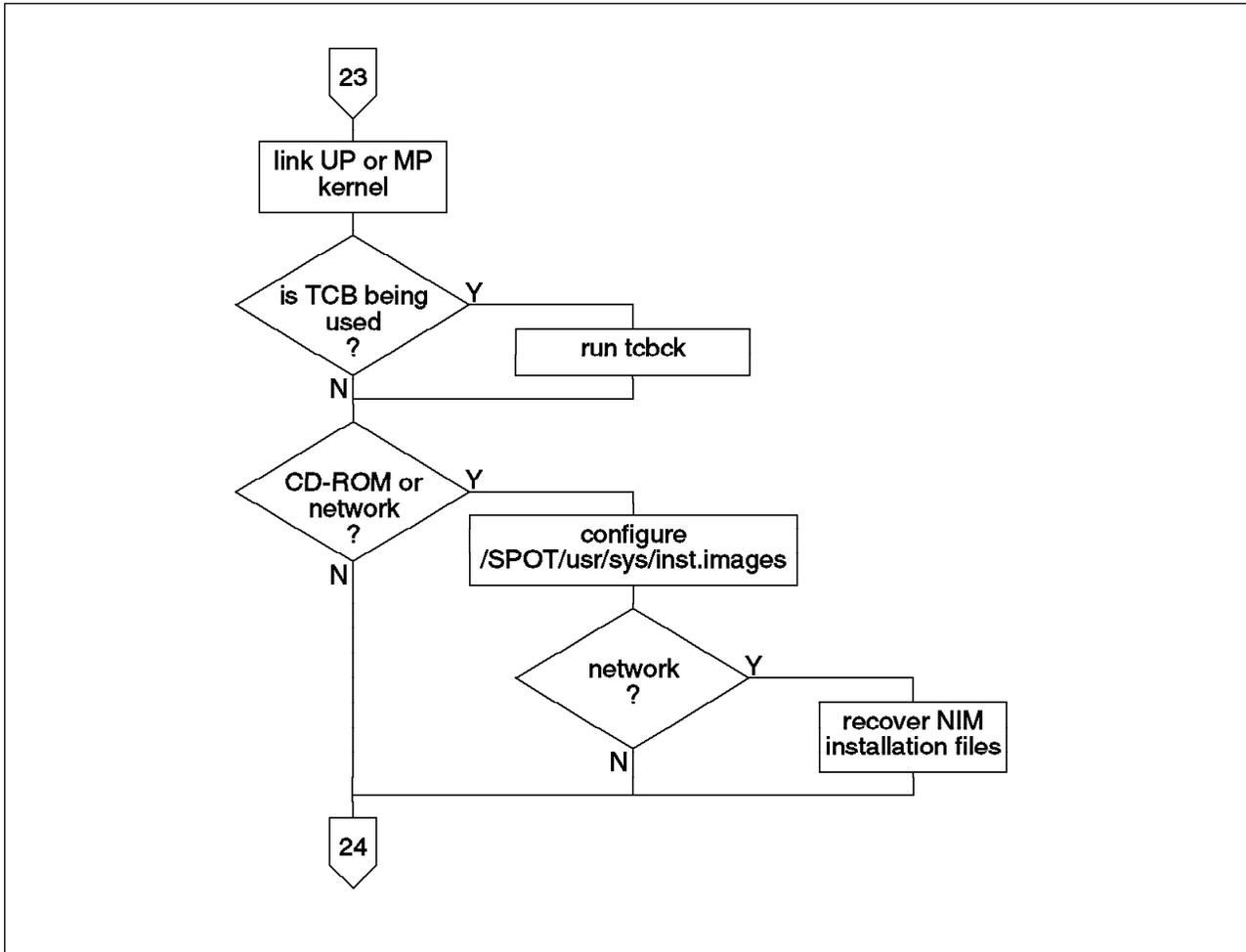
Figure 31. Post-Installation Processing for mksysb Installations.

First the unix kernel is linked to either the uni- or multi-processor kernel, as appropriate. If TCB is to be used on the system, a TCB check is now run. If the system is being installed from a network or CD, the installation directory /SPOT/usr/sys/inst.images is configured as an installation device. Also, if a network is being used, the program `c_bosinst_env` is now run again to set up the required files in the disk file systems. The `Change_Status` function is then called several times to jump the percentage indicator several places as the mksysb includes several fewer steps than other installation methods.

3.4.13.2 Post-Installation for Product Tape Installations

This much longer procedure for post-installation processing in migration, preservation and overwrite installations is shown in Figure 32 on page 131 to Figure 37 on page 138.
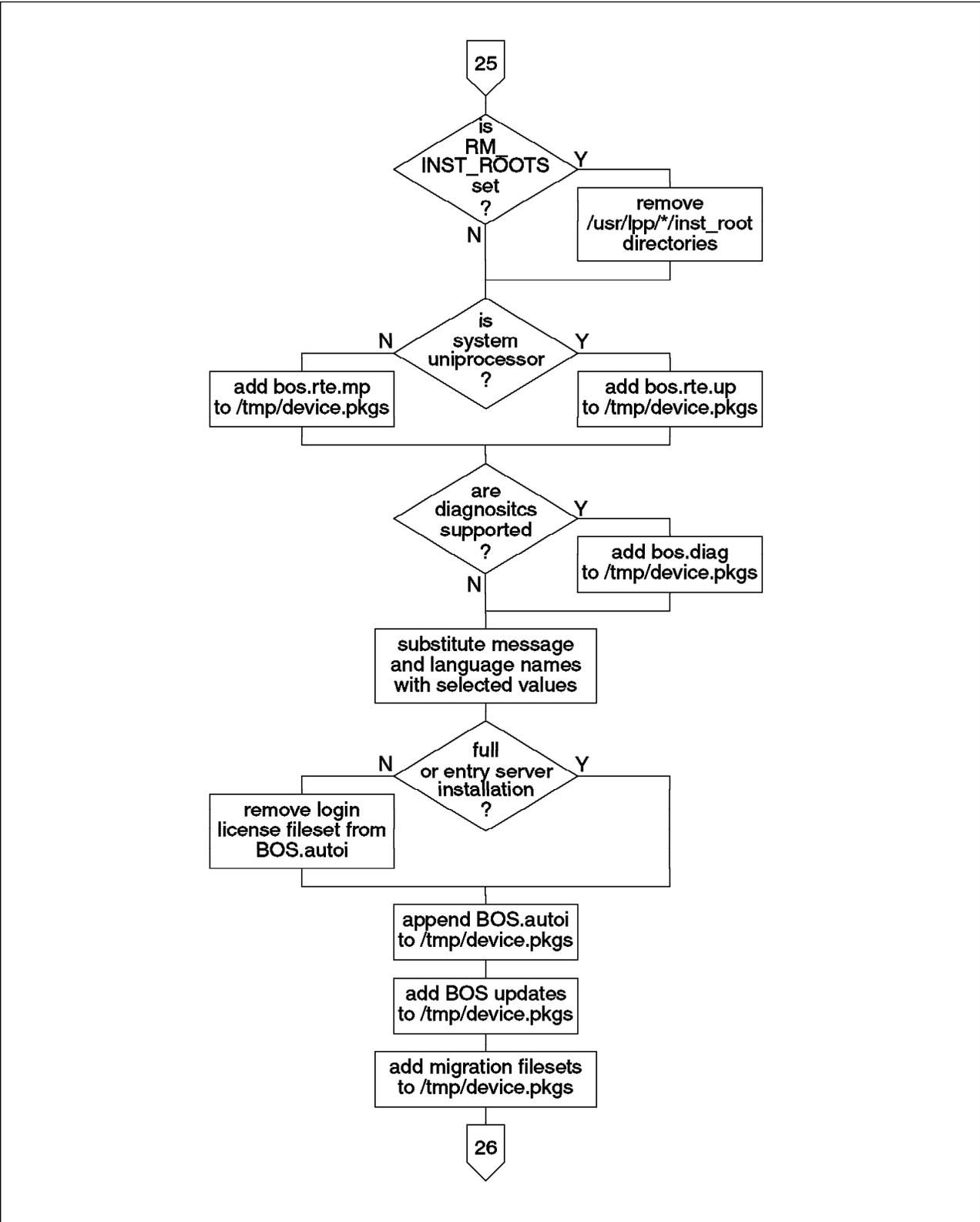
Figure 32. Post-Installation Processing Product Tape Installations - Part A.

The process starts by considering the RM_INST_ROOTS variable from the bosinst.data file. If this variable is set, the `/usr/lib/instl/inurid` shell script is used to delete the inst_root directory for each LPP. These directories are needed
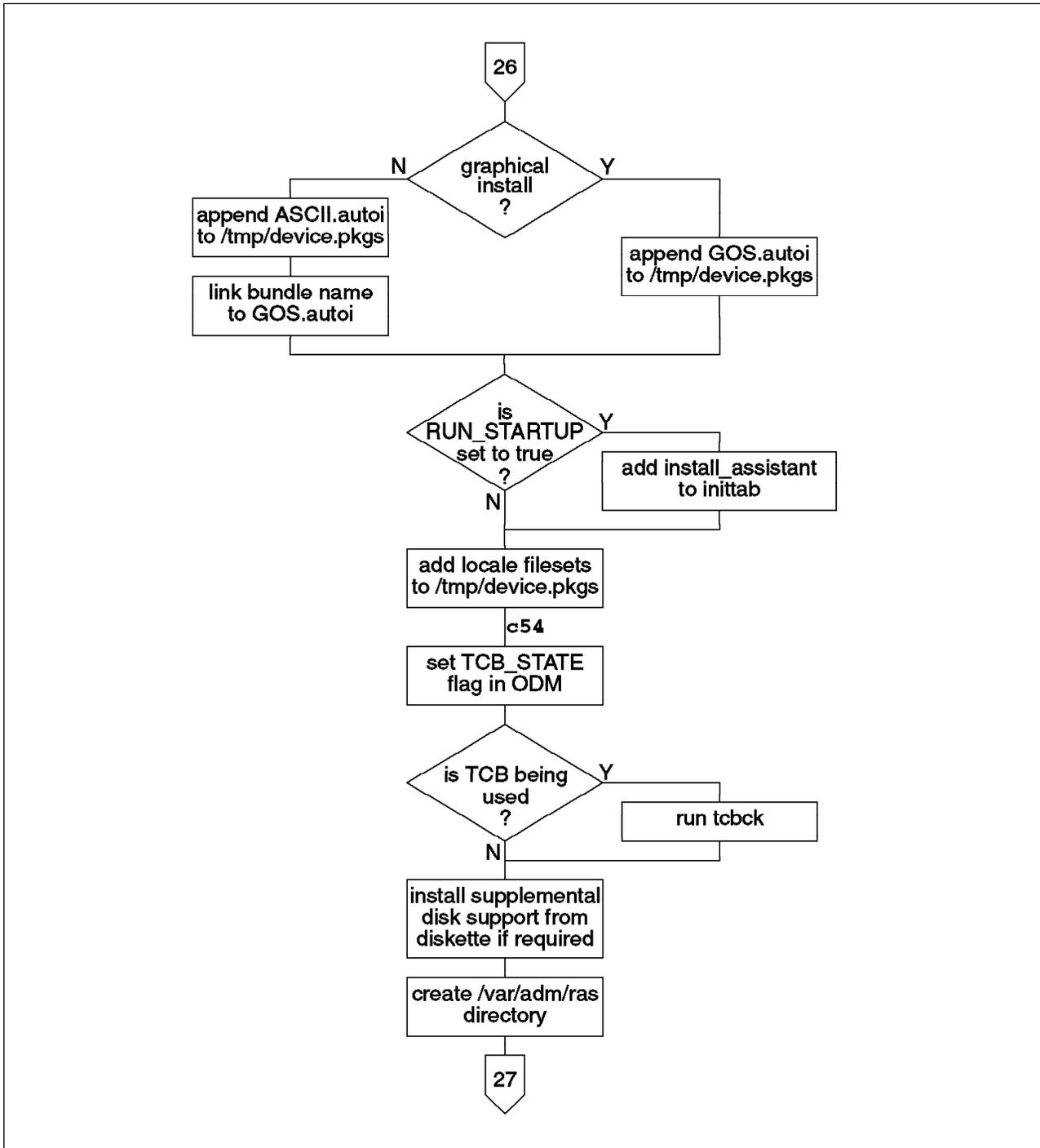
Figure 33. Post-Installation Processing Product Tape Installations - Part B. Bold numbers show LED display values.

only if the system is to be a network installation server, and disk space can be saved by deleting these areas.

The script then starts compiling a list in the file /tmp/device.pkgs in the RAM file system of filesets that must be installed. To begin with, either the uni- or multi-processor runtime environments are added to the list, depending upon the system architecture. If the system supports diagnostics, the diagnostics fileset will also be added. Then the selected locale information is edited into the appropriate positions in the /tmp/BOS.autoi file which is appended to our list. If the installation

type is not full and the system is not an entry server, the login license control fileset (bos.sysmgt.loginlic) is removed from the list. Thus the licensing fileset will only be installed for full (server) installations or on entry servers. The script then checks the installation media and adds any available BOS updates to the list. If a migration installation is being performed, the list of filesets determined byt the migration processing is appended. Similarly, if the graphics support is to be installed, these files are also listed. An additional set of locale specific files are also added to the list.

Meanwhile, if the RUN_STARTUP variable in the bosinst.data file is set to yes, the install assistant is added to the inittab to be run on the first reboot. If the Trusted Computing Base is being used, the TCB flag will be set in the ODM, and another TCB check will be run.

The script now tests to see if any supplemental disks have been chosen as target disks. A supplemental disk is one where the device drivers for the particular disk device are not included in the boot and installation files. In this case, the script expects to install the additional device drivers from diskette. At this point it attempts to restore a file ./signature from the diskette. If the file exists and contains the string "target", the system will install the device drivers from the diskette.

The script now verifies that the directory /var/adm/ras exists on the newly installed system. This directory is used for storing various log files that are kept both during the installation and during normal system operations. For more information on the installation logs, see 1.5.3, "Retrieving Installation Messages" on page 42.

In Figure 34 on page 134 we see the actual installation of the additional packages from tape. First `bi_main` uses the command `/usr/lib/instl/automsg` to list all of the message files on the tape that correspond to our selected language and cultural convention settings. This list is appended to the end /tmp/device.pkgs. This ensures that for each fileset being installed, the corresponding message fileset will be installed. Any message filesets that are not required will fail in requisite checking and thus not be installed. Note that if the language and cultural conventions are set to **C**, no message or locale filesets will be installed. The script then attempts to install the filesets listed in the /tmp/device.pkgs file. This is where the additional OPPs and LPPs are installed on the system. For more information on the processing performed for each LPP, see 3.6, "Program Product Migration" on page 151. The most likely reason for the installation failing at this time is a lack of space, since the installation process requires more space available than will eventually be used by the packages. If the installation of all filesets fails, the system will instead try to install just the minimum set of filesets plus graphics support if appropriate. If this also fails, just the minimum list of filesets will be attempted. If the installation still fails, `bi_main` gives up and displays an error message. If the installation has been successful, the script checks to see if the BUNDLES variable in bosinst.data specifies any additional bundles of software to be installed. If additional bundles are specified, `automsg` again appends the list of message files before installing the listed bundles.
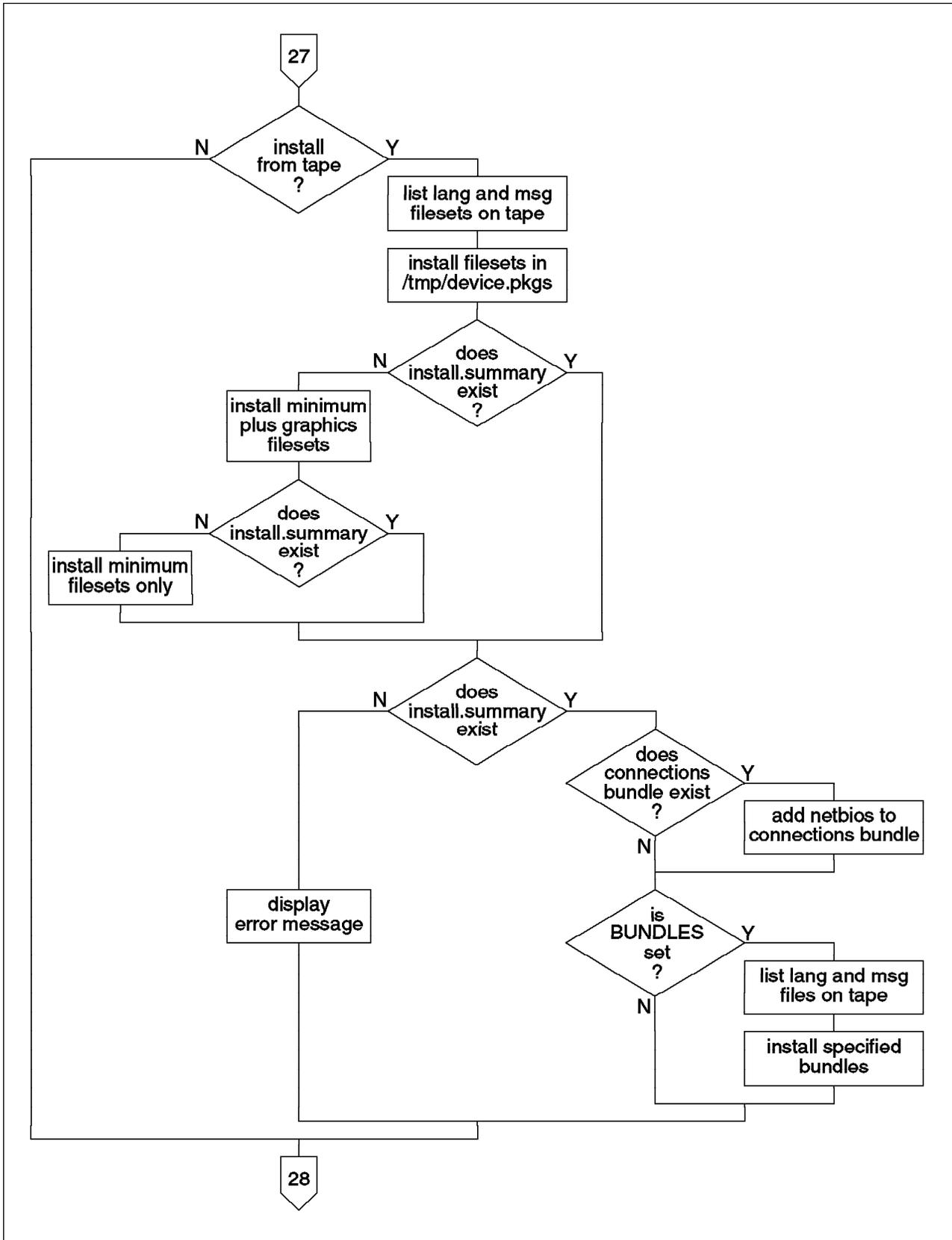
Figure 34. Post-Installation Processing Product Tape Installations - Part C.
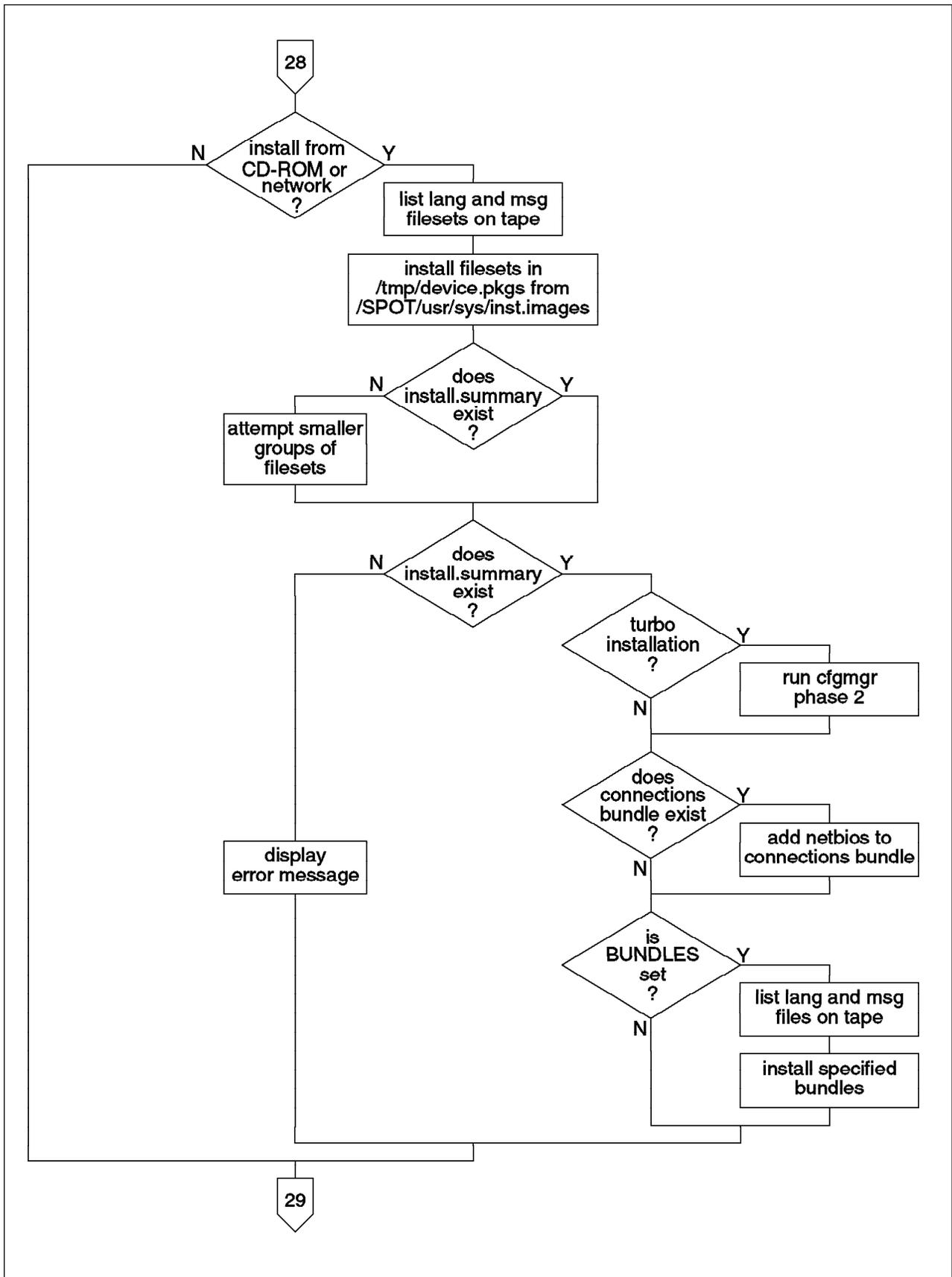
Figure 35. Post-Installation Processing Product Tape Installations - Part D.

Figure 35 shows the identical process for network or CD-ROM installations. In this case, there is one additional step. If a turbo installation was run, phase two of device configuration is run again to ensure that any required kernel extensions are loaded before calling the migration pre-installation routine.

Next, in Figure 36 on page 137, the script creates the installation bundle files by linking the <bundle name>.bnd file names to the <bundle name>.def files that are shipped as part of the program distribution. In the case of a client system, a client bundle is not created because all of the client files are installed by default.

If the installation is by network from a NIM SPOT, the script next runs the `installp` `-X0r` command. When the files are copied from the SPOT they contain the usr part of bos and possibly other products, and includes the ODM databases with the Software Vital Product Data for these products. This `installp` command will now install the corresponding root parts of these products. The unix kernel is then linked to either the UP or SMP kernel, and `c_bosinst_env` is again run to ensure that the setup is correct to enable NIM processing to continue.

At this stage `bi_main` calls `cfgmgr`, specifying either the tape device or the installation directory for CD-ROM or network installations. This tells `cfgmgr` that it should install device support for any devices that it discovers.

As shown in Figure 37 on page 138, the `bi_main` script then verifies that the International Language Support (ILS) filesets have installed correctly before setting the locale related variables in the /etc/environment file for future operations.

In the case of an entry server installation, the default number of login licenses is then set to 16 users.
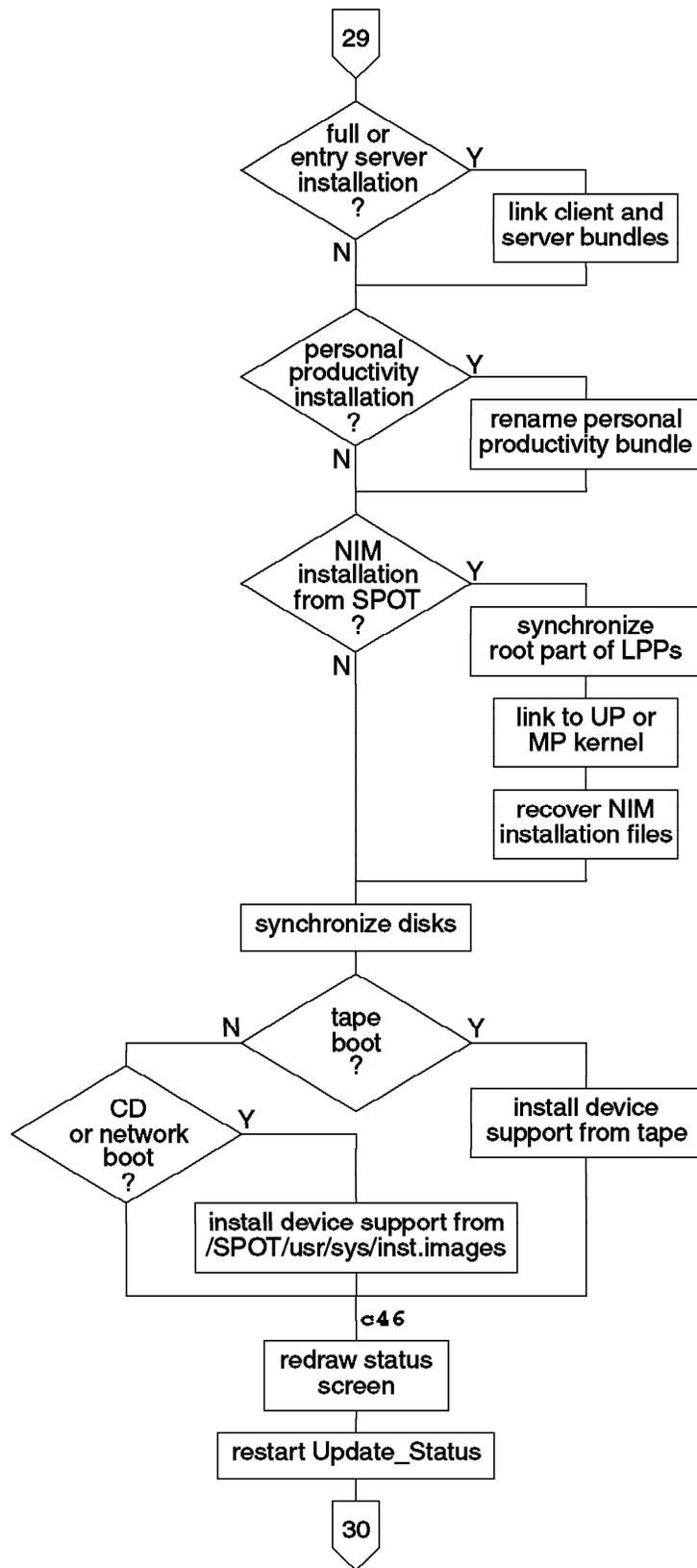
Figure 36. Post-Installation Processing Product Tape Installations - Part E. Bold numbers show LED display values.
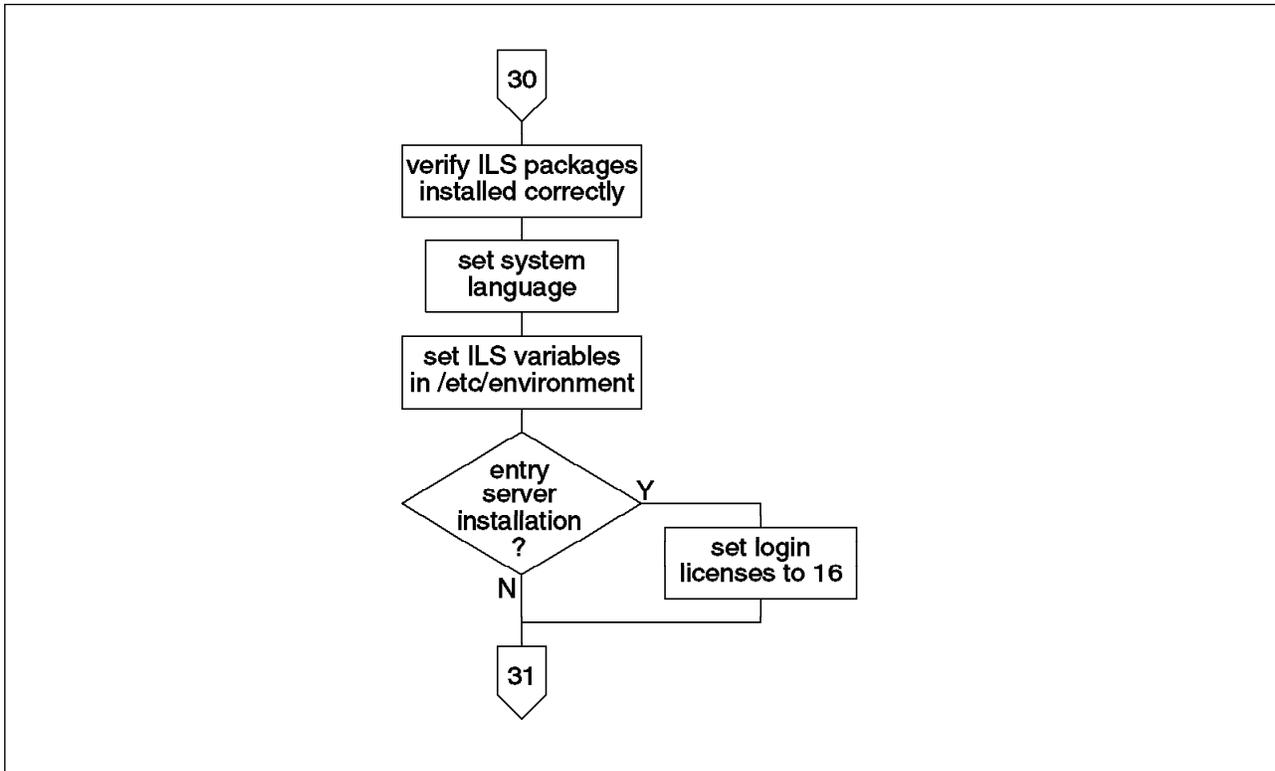
Figure 37. Post-Installation Processing Product Tape Installations - Part F.

3.4.13.3 More Post-Installation Processing

The two cases of mksysb and product tape installations now rejoin to complete the installation post-processing. The steps shown in Figure 38 on page 139 and Figure 39 on page 140 are performed in all cases.

First, if TCB has been installed, the program `/usr/lpp/bos/mksysck` called to initialize the system check configuration database with any device special files created by the installation process.

Next, if the dump space is not defined, it is set to the paging space. In a migration or preservation installation from AIX Version 3.2, this would be the case if the dump space in the old system had been the standard hd7 dump device. In this case the dump device was removed earlier. The /etc/swapspaces file is also initialized to contain the names of the paging spaces.

If the ODM was earlier preserved in /tmp/bos/objrepos, we now add entries to the /etc/firstboot script to recover the device attributes using the `/usr/lpp/bosinst/rda` command. The `/usr/lpp/bosinst/modprtdevs` command is called to modify the Version 3.1 or 3.2 printer definitions into the format required for AIX Version 4.1.

In the case of a mksysb installation, the script now checks the tape blocksize setting. If the block size is set to be less than 512, commands are added to /etc/firstboot to change it to its original setting.

In the case of migration and preservation installations, it is now time to import any non-rootvg volume groups that were previously defined on the system. Earlier the file /tmp/targetvgs was built, listing the VGID, operating system level, and other disk information. Also, the file /tmp/bos/oldvgs/.oldvgs was created, which lists the
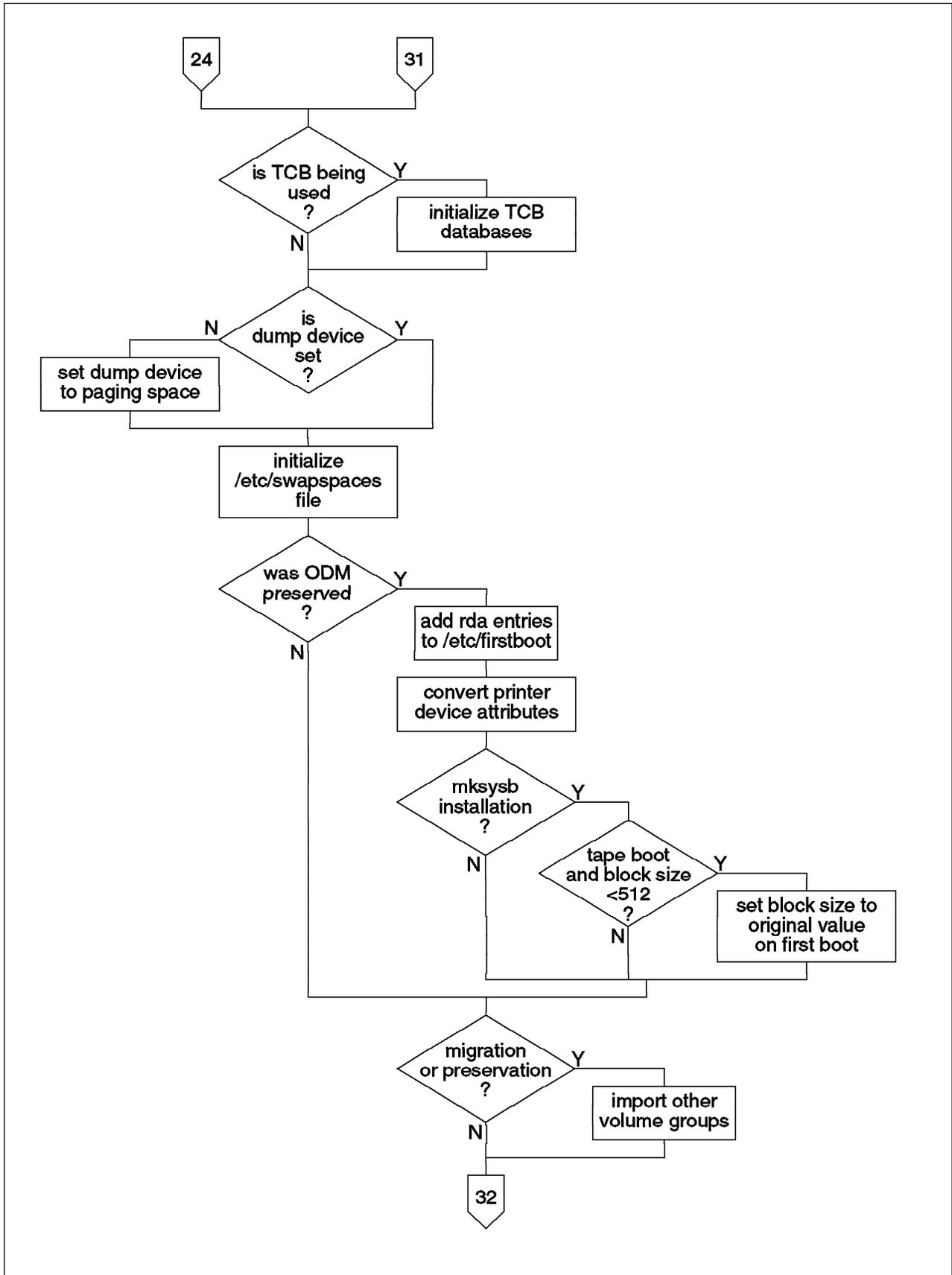
Figure 38. More Post-Installation Processing - Part A.
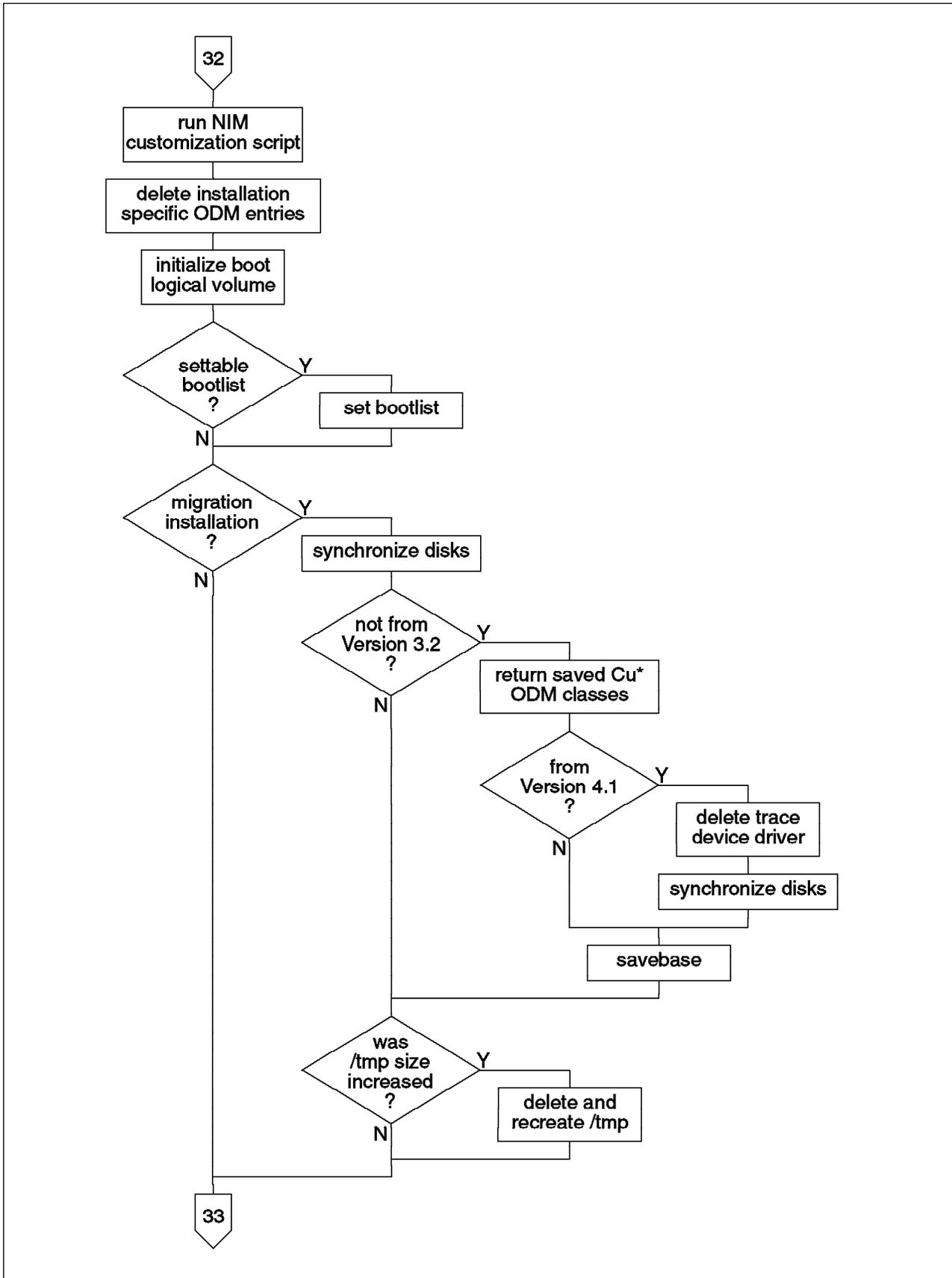
Figure 39. More Post-Installation Processing - Part B.

VGID and the VG name. These two files are now used to import the existing volume groups. For each line in the .oldvgs file, the script finds a corresponding disk in the targetvgs file and imports the volume group from that disk. In an `importvg` operation, the information regarding the volume group is copied from the Volume Group Descriptor Area (VGDA) on the disk to the ODM. There are two additional volume group settings whose values are stored only in the ODM:

- Activate volume group automatically at system restart

  This setting determines if the volume group should be activated, or varied on, as the system boots. The default value for this setting is **yes**.

- A quorum of disks required to keep the volume group on-line

  Determines if over 50% of the volume group VGDAs must be available for continued access to the volume group

Since these settings are stored only in the ODM and not in the VGDA, they are not preserved through the preservation or migration. Thus when the existing volume groups are imported, they will be set to the default values listed above. If the values before migration were set differently, these changes must be made manually.

If the import of the other volume groups fail, the most likely reason is a lack of space in the root file system. If there is less than two kilobytes free in /, the script will increase / by one partition and try to import the volume groups again.

In Figure 39 on page 140, the script now checks to see if the NIM_CUSTOM variable was set in the NIM information file. This variable names a NIM customization file, and by default is set to the script `/SPOT/usr/lpp/bos.sysmgt/nim/methods/c_script`. If defined, the script will now be run.

Next the script deletes some installation specific entries from the ODM before running the `bosboot` command to initialize the boot logical volume for subsequent reboots. If the system supports a settable bootlist, this will also be set at this time to specify the newly installed system.

Finally, a few last migration related steps. If the size of the /tmp file system was increased during the migration, and the additional space is no longer required, the file system is deleted and recreated at the original size. If the migration is from a previous modification level of AIX Version 4.1 to Version 4.1.4, the customized ODM classes are copied back to their directory to be used as is. If the migration was from AIX Version 4.1 specifically, the tracing devices are deleted.

After running `sync` to synchronize the disks one more time, the post-installation process is complete.

3.4.14 Run Customization

The function Run_Customization, shown in Figure 40 on page 142 has little to do. It simply checks to see if the variable CUSTOMIZATION_FILE in the bosinst.data file is set to indicate a user customization script. If not, it checks the variable BOSINST_FILE in the image.data file, which can also define a customization script. If either variable is set, the specified script will be copied to the disk file systems and executed.
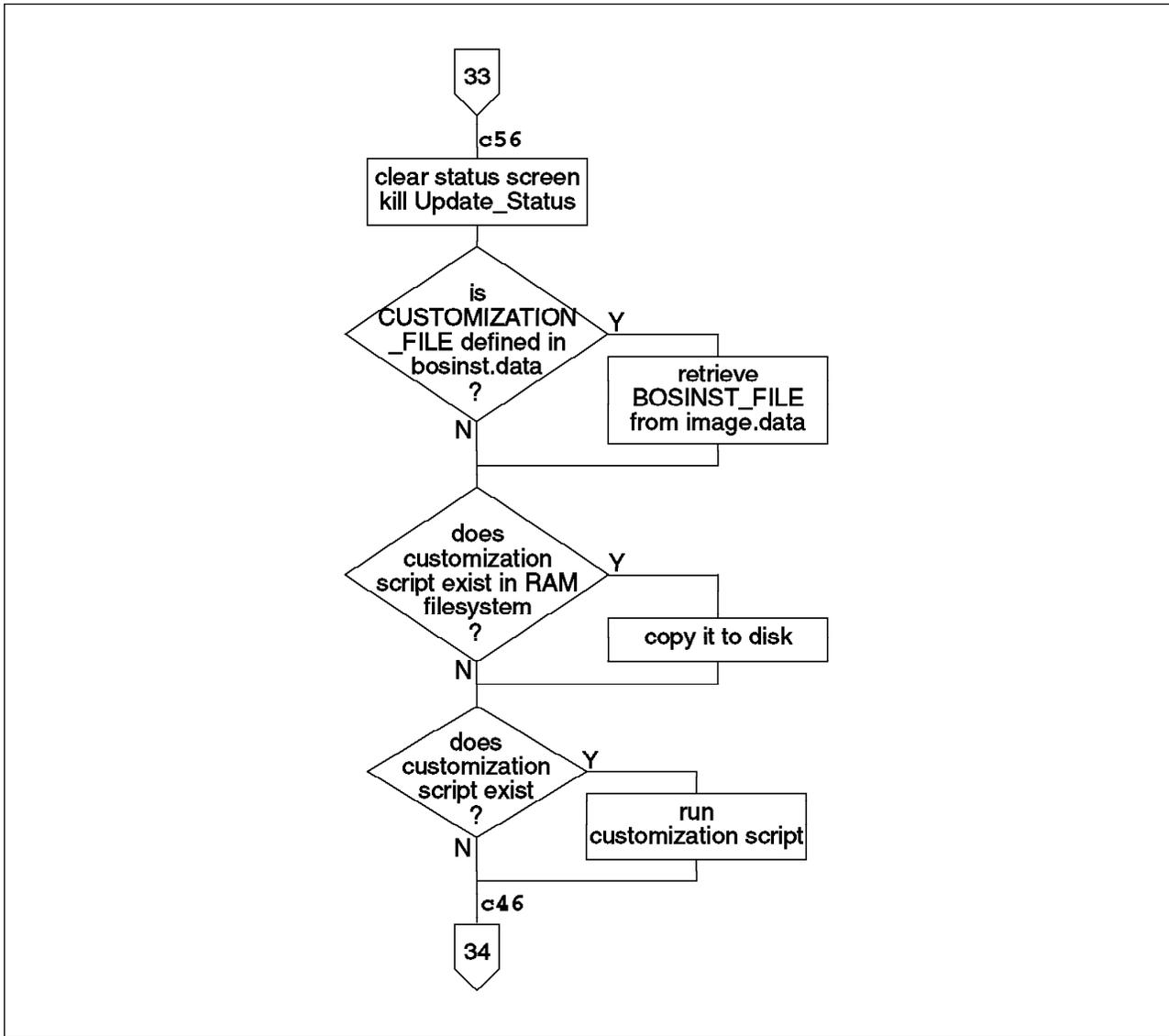
Figure 40. Running Customization. Bold numbers show LED display values.

3.4.15 Finish

This function performs the cleanup operations and informs the user of the results of their installation. It is shown in Figure 41 on page 144.

First the percentage complete value is set to 100%. Next, the installation logs are copied from the RAM file system to disk to ensure that they will be available to the user after the system is rebooted.

In the case of a migration from a lower modification level of AIX Version 4.1, the device special files in the /dev directory are now deleted, and the original /dev files are replaced. This is permitted in a 4.1.x to 4.1.4 migration since the device and ODM entries are compatible.

If the installation was unsuccessful, we must ensure that the user sees the error messages. Thus the status will be displayed, and the system will wait for the user to press enter before rebooting. Likewise, if the key switch, where present, has not been turned to the normal position, the user is prompted to do so before rebooting. In systems where setting the bootlist is not supported, it is necessary for the user to remove the installation media to prevent the system from booting from the installation media again. At present, the RISC System/6000 model 40P is the only model that does can not set the bootlist.

The system is then rebooted—the installation process is complete.
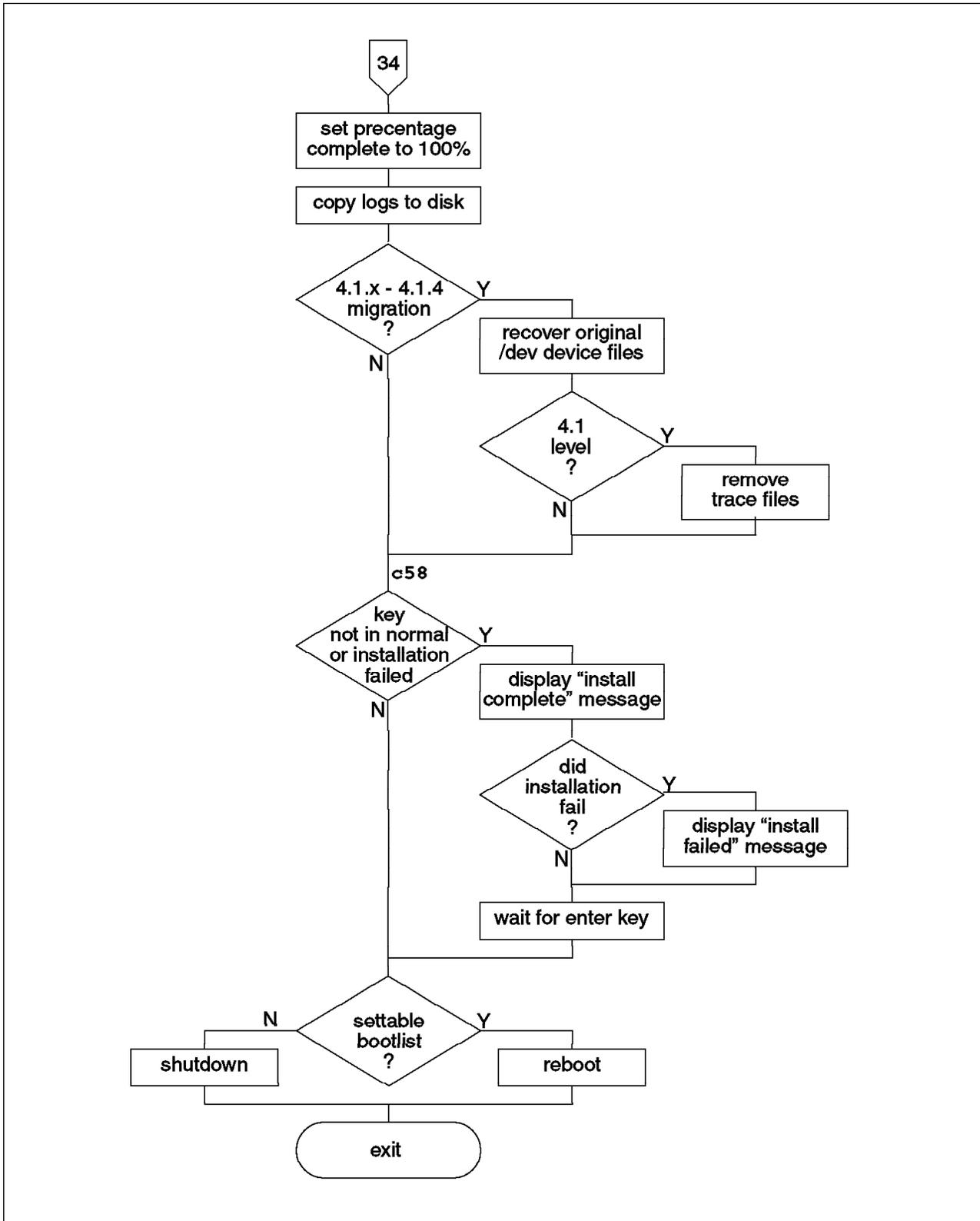
Figure 41. Finishing the Installation. Bold numbers show LED display values.

3.5 Migration Specific Processing

As we have seen above, there are many variations to the installation process depending upon the type of installation being performed. The most complex case is the migration installation, where there are procedures for removing the old files, determining the new filesets to install and replacing or migrating configuration information.

The migration processing is mostly performed in the shell scripts `bos.rte.pre_i` and `bos.rte.post_i`. These scripts are detailed in the sections below. As well as these two scripts, the migration process depends upon the following files:

- bos.rte.root.rmlist

  A simple list, one file per line, of files in the root part of filesets that must be removed during the migration.

- bos.rte.usr.rmlist

  A simple list, one file per line, of files in the usr part of filesets that must be removed during the migration.

- incompat.pkgs

  A list of AIX Version 3.2 filesets that are incompatible with AIX Version 4.1 and one or more Version 4.1 filesets that should be installed to replace that fileset. Each V3.2 fileset can be listed more than once to allow the listing of many corresponding V4.1 filesets.

- bos.rte.cfgfiles

  A list of configuration files that must be handled in some way during the migration. The files will be saved during pre-installation processing and replaced or processed during post-installation. Each line lists the file name and path, the action that must be taken, and, in some cases, the name of the routine used to merge the file with the new version.

  The actions to be performed for each configuration file are identified by keywords taken from the following list:

  preserve The file should be moved back to its original position after the installation.

  v4preserve The file should be preserved (or moved back to its original position) after a V4.1 migration, but for a V3.2 migration it should be left saved under /tmp/bos.

  user_merge The file is left saved under /tmp/bos. If there are any settings in the file that are required, the user must manually move these settings into the new version of the file.

  hold_new The old and new versions of the file will be swapped. The old version will become active. The new version will be saved under /tmp/bos so that any required settings from the new V4.1 version of the file can be merged into the old version if required.

  auto_merge The file is moved back to its original position, however in most cases a merge method will also be specified with auto_merge files. The merge method will be run first to merge any settings or new requirements from the V4.1 version of the file with any user or site

specific settings from the V3.2 file. The resulting combined file is then moved to the active position.

other   This is handled as a user_merge and left saved under /tmp/bos.

These files are used as input to the script to specify either files to be removed, or configuration files that must be preserved.

In many cases the migration scripts refer to two separate migration cases. The most common situation is the migration from AIX Version 3.2 to Version 4.1. We will refer to this as a V3.2 migration. The other case considers the migration from AIX Version 4.1.0 to a later modification level of AIX Version 4.1. For simplicity we will refer to this as a V4.1 migration.

3.5.1 Migration Pre-Installation Processing

The migration pre-installation processing script `bos.rte.pre_i` is called by the `BI_Copy_Files` function of `bi_main` (see 3.4.8.1, "Prepare Target Disks for Preservation or Migration" on page 118 and Figure 23 on page 119). It is either copied to the disk from the RAM file system or extracted from the liblpp.a library along with the files listed above.

At the time the script executes, the disk file systems will be mounted on the /mnt directory. For example, hd4 will be mounted on /mnt, hd2 will be mounted on /mnt/usr. In addition, since the `lslpp` command, which will be used to determine the installed software, does not use the ODMPATH or ODMDIR environmental variables to specify the ODM location, the disk ODM directories are be mounted directly over the ODM directories in the RAM file system. For example, the disk directory /mnt/etc/objrepos is mounted over /etc/objrepos in the RAM file system.

The steps in the migration pre-installation process are shown in Figure 42 on page 147 and Figure 43 on page 148.

The case of a V3.2 migration is handled first. This process starts by building a list of filesets to remove from the system. These are filesets whose function is no longer supported under V4.1, or filesets that will not be automatically installed by the migration process. It builds this list by comparing the list of currently installed products against a list that is coded in the `bos.rte.pre_i` shell script. It also checks for some conditional filesets, such as data link control filesets (dlc). The dlc filesets will only be installed if the corresponding V3.2 dlc fileset was both installed and configured.

Next the script builds a list of files that will later be checked to see that they have been moved or deleted. This list is obtained directly from the bos.rte.usr.rmlist and bos.rte.root.rmlist files provided.

The script then considers the file incompat.pkgs. This file lists the V3.2 filesets that are incompatible with V4.1, and for each V3.2 fileset, lists one or more V4.1 filesets that should be installed to replace or upgrade it. This file is compared with the installed V3.2 filesets to generate a list of optional filesets that must be installed as replacements or upgrades of V3.2 filesets. In addition, the script uses the `lslpp` command to add the names of all files in the filesets chosen to the bos.rte.usr.rmlist and bos.rte.root.rmlist files.
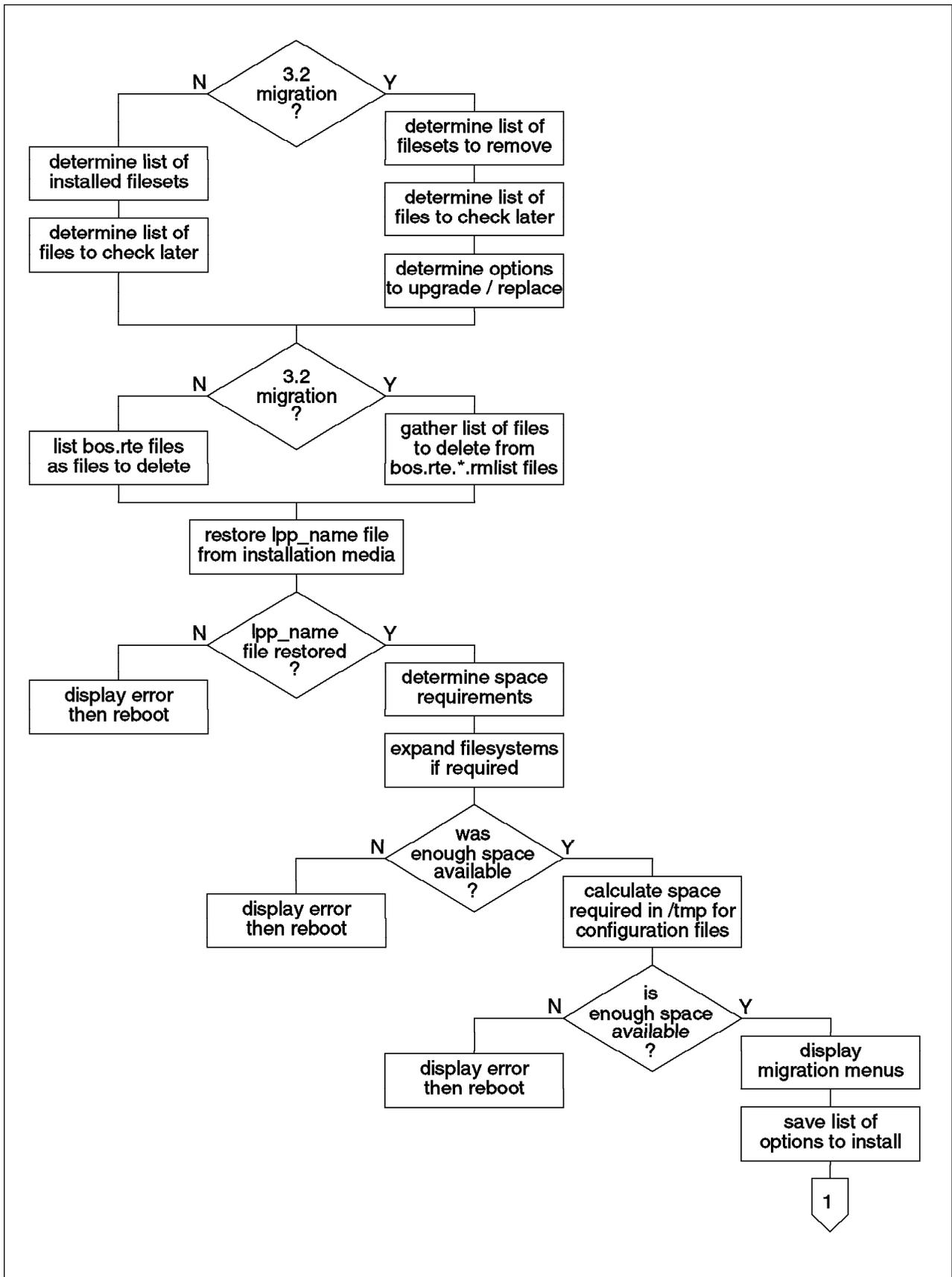
Figure 42. BOS Migration Pre-Installation Processing - Part A.
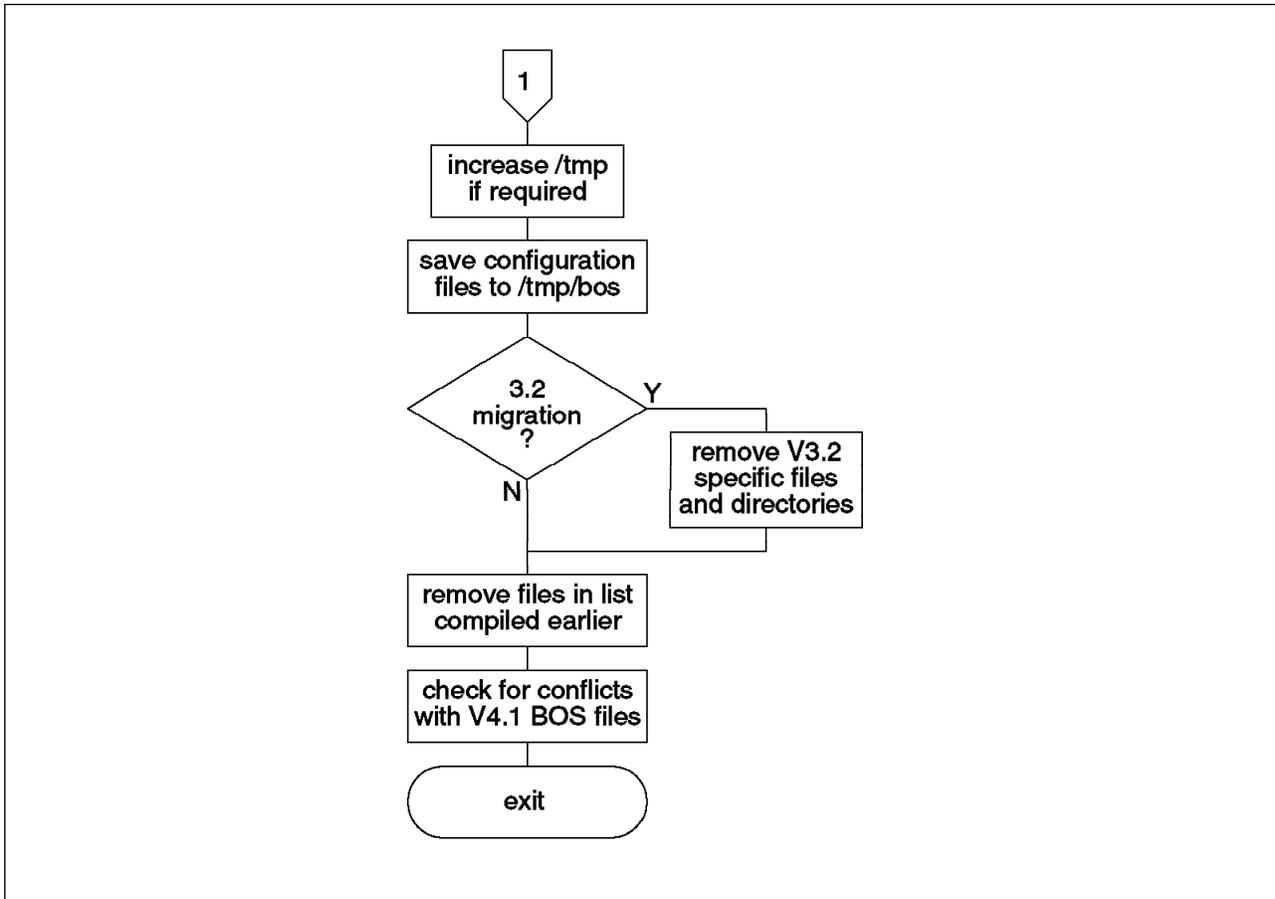
Figure 43. BOS Migration Pre-Installation Processing - Part B.

In the case of a V4.1 migration, the script simply builds a list of filesets that are currently installed, along with a list of files to later be checked. The latter list is again taken from the bos.rte.usr.rmlist and bos.rte.root.rmlist files.

Next `bos.rte.pre_i` gathers a list of files to be deleted. In the case of a V3.2 migration this is simply a combination of the bos.rte.usr.rmlist and bos.rte.root.rmlist files, slightly reformatted. In the case of a V4.1 migration, the script simply lists the files in the existing bos.rte image as the files to be deleted.

The `bos.rte.pre_i` script then begins to look at the space requirements for the migration, considering such factors as:

- The space required for new V4.1 files in each directory, taken from the lpp_name file

- The V3.2 directories that will be deleted, and the space that is taken by the files in these directories

- Additional miscellaneous files that will be deleted

The program `/usr/lpp/bosinst/adjfs` is then called with all of this information to calculate if the space is sufficient, and to expand the file systems if required. If sufficient disk space is not available, an error message is displayed and the system reboots.

Next the system verifies that there is sufficient disk space in the disk /tmp directory (currently mounted on /mnt/tmp) to perform the migration. The /tmp directory must be used to store the configuration files during the migration. Also, the files that cannot be automatically merged with or moved over their V4.1 counterparts will be left in the /tmp directory to be examined by the user. The /tmp file system must also have sufficient space to allow a new boot sector to be generated at the end of the migration. The script compares the space that is required in /tmp against the space that is available in /tmp and the additional physical partitions that are not allocated to a volume group. At this stage the calculation does not take into account any space that will be saved by removing the dump space.

The system then displays the migration menus, as shown in 1.4, "Migrating AIX" on page 33. They includes an option to exit and reboot without migration.

If the user chooses to continue with the migration, the script will save the list that was determined earlier of optional products to install ready to be installed with the other packages by `bi_main`. If required, the script will then increase the size of /tmp and save the configuration files listed in the bos.rte.cfgfiles file under the directory tree /mnt/tmp/bos/. The files are saved in their regular structure. For example, the /etc/inittab file will be saved under /mnt/tmp/bos/etc/inittab while /var/spool/cron/crontabs/root will be stored under /mnt/tmp/bos/var/spool/cron/crontabs/root.

It is now time for `bos.rte.pre_i` to remove the V3.2 related files and directories. This includes such files as:

- PTF specific ODM entries
- ODM entries for removed packages
- files for devices, device drivers, microcode and device special files (in /dev directory)
- diagnostics
- package files such as /usr/lpp/bosext1 and /usr/lpp/bosnet
- obsolete printer and queueing files
- obsolete graphics databases

Finally the script deletes the actual BOS files according to the list that was generated earlier. It then looks at the list of files to check that was compiled earlier. Any files in this list that still exist will conflict with files that are to be installed as part of the BOS. Thus any remaining files or directories will be moved to a new name made from the old name, the word "save" and the process ID of the migration program. If the file cannot be moved, it will be deleted. The names of any files moved or deleted will be stored and shown to the user during post-installation processing.

3.5.2 Migration Post-Installation Processing

The migration post-installation processing script `bos.rte.post_i` is called by the `Call_Merge_Methods` function of `bi_main` after being extracted from the liblpp.a library (see 3.4.13, "Post-Installation Processing" on page 128 and Figure 30 on page 129). At this stage of the installation process the disk file systems have been mounted over the RAM file system. For example, hd4 is mounted on / and hd2 is mounted on /usr. The configuration files that were saved by the pre-installation

routine are stored under the /tmp/bos directory tree. The migration post-installation process is shown in Figure 44 on page 150.

```
            ┌─────────────────┐
            │ display moved or│
            │  deleted files  │
            └────────┬────────┘
                     │
                  ╱──┴──╲
                ╱  3.2    ╲       Y
               ╱ migration  ╲─────────────┐
               ╲     ?      ╱             │
                ╲─────────╱      ┌────────┴────────┐
                 N │              │ remove remaining│
                   │              │ PTF information │
                   │              │    from ODM     │
                   │              └────────┬────────┘
                   │                       │
                   │              ┌────────┴────────┐
                   │              │   remove V3.2    │
                   │              │  prime messages  │
                   │              └────────┬────────┘
                   │                       │
                   │              ┌────────┴────────┐
                   │              │  remove SWVPD    │
                   │              │for deleted filesets│
                   │              └────────┬────────┘
                   │                       │
                   │                    ╱──┴──╲
                   │                  ╱   is    ╲    Y
                   │                 ╱root filesystem╲──────┐
                   │                 ╲ only 4 MB    ╱       │
                   │                  ╲    ?      ╱   ┌─────┴─────┐
                   │                   ╲───────╱     │increase size of│
                   │                    N │          │root filesystem │
                   │                      │          │    by 4 MB      │
                   │                      │          └─────┬─────┘
                   │                      └────────────────┘
                   │
          ┌────────┴────────┐
          │  process saved   │
          │configuration files│
          └────────┬────────┘
                   │
          ┌────────┴────────┐
          │   modify SRC     │
          │  configuration   │
          └────────┬────────┘
                   │
          ┌────────┴────────┐
          │  execute install │
          │assistant migration│
          │      script       │
          └────────┬────────┘
                   │
          ┌────────┴────────┐
          │determine upgraded│
          │options to reinstall│
          └────────┬────────┘
                   │
              ╭────┴────╮
              │   exit   │
              ╰─────────╯
```

Figure 44. BOS Migration Post-Installation Processing.

The script begins by displaying the names of any files or directories that had to be moved or deleted to avoid clashes with V4.1 BOS files.

If a V3.2 migration is being performed, the script now deletes any remaining PTF information from the ODM before removing any obsolete prime message catalogues. It also deletes the Software Vital Product Data (SWVPD) from the ODM for any filesets whose files were removed during pre-installation processing.

Next, if the root (/) file system is just 4 MB in size, there is a high probability that there will not be sufficient inodes available. To prevent this problem, the root file system is increased by one physical partition.

This is followed by the processing of the configuration files that were saved during the pre-installation processing. The /tmp/bos/bos.rte.cfgfiles files is scanned. For each file, the action to be performed, and in some cases a merge method, are read from the file. The merge method is executed, then the file is either copied back to its original position or left in the save directories according to the rules listed in 3.5, "Migration Specific Processing" on page 145.

The System Resource Controller (SRC) configuration is now modified. Many old SRC entries are deleted. These will be re-added if appropriate as the corresponding products are installed. Also, the SRC entries for qdaemon, writesrv and lpd are changed to reflect the new locations of their executables.

If the install assistant included its own migration script called `/usr/lib/assist/assist_migrate`, this script is now executed.

Finally the script considers each installed option package (which were listed during pre-installation processing) and checks to see if a superseding fileset is available on the installation media. If so, that fileset is listed to be installed during the installation of additional packages in the `Post_Installation` routine of `bi_main`.

3.6 Program Product Migration

Up until this stage the whole chapter has basically covered the migration of the base operating system, and with good reason. This is by far the most important and complex part of the overall migration process. However, almost every system installations will also have some Optional Program Products (OPPs) and possibly some Licensed Program Products (LPPs) installed. Each of these products has its own migration requirements. In most cases there will be configuration files to preserve or merge. In some cases the name of the package may have changed, or the V3.2 package may not map to new V4.1 packages on a one-to-one basis. In all migration cases there will be old files to delete or replace, and new files with potentially different space requirements to install. This chapter deals with the processing required to perform these steps.

While the BOS installation has long and convoluted shell scripts specifically coded to perform the many steps involved in operating system migration, this luxury would be a huge burden to maintain for every available OPP or LPP. Thus a standardized, generic approach to OPP and LPP migration is required. The approach uses the standard AIX installation command `installp` and a set of configuration files and scripts that mimic the general steps performed during the BOS migration, but on a much smaller scale.

Before considering the process used in the migration of OPPs and LPPs, it helps greatly if we first look at the files that make up an installable package, with particular emphasis on those files used by the installation and migration process. We will generally not discuss the packaging requirements of software updates as they are not particularly relevant to our discussion of migration from AIX Version 3.2 to Version 4.1.

Each software product is packaged in a backup format file that contains the following files:

- ./lpp_name

- ./usr/lpp/<package name>/liblpp.a

- The files that comprise the actual contents of the software package, backed up relative to the root directory.

In addition, if the software package includes a root part it will include the following files:

- ./usr/lpp/<package name>/inst_root/liblpp.a

- The files that make up the root part of the software product, backed up relative to the directory ./usr/lpp/<package name>/inst_root/.

It is interesting to note that files that will eventually be placed in the root file system directories such as /etc or /sbin will be first restored in the /usr file system. The installation process will move the files to their eventual positions. This allows the root part of a fileset to be installed from this directory to many different systems. This is the technique used when the /usr file system is being shared between diskless and dataless clients, and for consitency, the same technique is used on stand-alone systems.

If the software product has a share part it must be packaged in a separate backup file which will include the following files:

- ./lpp_name

- ./usr/share/lpp/<package name>/liblpp.a

- The files that are included in the share part of the product, backed up relative to the root directory.

3.6.1 lpp_name

This file contains information about the package and each fileset it contains. The file includes basic data such as the name of the package and the platform for which it is built, then a series of detailed stanzas for each fileset. The fileset stanzas include such information as:

- the fileset name

- the software level

- whether the fileset includes just the usr part, both usr and root parts or the share part

- a fileset description and comments

- requistite information, listing dependencies on other filesets

- size information, including a directory-by-directory listing of both the permanent and temporary disk space requirements for the fileset

- supersede information

For more information on the contents of the lpp_name file, see *InfoExplorer* or *AIX Version 4.1 General Programming Concepts: Writing and Debugging Programs*

3.6.2 liblpp.a

This file, known as the installation control library file, contains the shell scripts and data files that will control the installation and migration of the filesets in the package. liblpp.a is an AIX archive file created with the ar command. To list the files in the archive, use the command:

```
ar -t liblpp.a
```

To extract all files from the archive, use:

```
ar -x liblpp.a
```

To extract specific files, specify their file names on the command line as follows:

```
ar -x liblpp.a file1 file2 file3 ...
```

There are many files that can potentially be included in the archive. In the listing below, we show only those most relevant to a migration from AIX Version 3.2 to AIX Version 4.1. In the following list, <fileset> indicates the name of the fileset, such as bos.net.tcp.server.

<fileset>.namelist This file is used in cases where the name of the fileset has changed between AIX V3.2 and V4.1. It lists the names of obsolete filesets that once included any of the files that are now included in this fileset.

<fileset>.al Apply List—The list of files that will be installed by this fileset.

<fileset>.installed_list This file is created by the installp command if it is found that the fileset or some part of it is already installed on the system. It lists the name of the fileset, or any other names that appear in the <fileset>.namelist file that are presently installed, and shows the level of each fileset.

<fileset>.cfgfiles A list of user configurable files for this fileset, showing the file name, a keyword indicating the action to perform. This file follows the same format as the bos.rte.cfgfiles file described in 3.5, "Migration Specific Processing" on page 145 with the exception that it does not specify merge methods on a file-by-file basis. It is assumed that any merging required will be performed by the <fileset>.post_i routine, or by programs called by it.

<fileset>.rm_inv This file is also used where the name of the fileset has changed between levels. It lists the path to any files that are part of the obsolete fileset and much be removed, but are not listed in the <fileset>.inventory as being a file that will be installed as part of the new fileset. This could be due to the fact that the file is stored in a new location in the new version of the fileset.

<fileset>.inventory The Software Vital Product Data (SWVPD) for the fileset, used to update the ODM databases. It contains one stanza for each file being installed as part of the fileset.

<fileset>.pre_rm This is a shell script that will be run (if present) before saving the configuration files and removing the old files from the obsolete version of the fileset.

<fileset>.pre_i This script will be run after removing the obsolete files, but before installing the current files.

<fileset>.post_i This script is run after installing the new files.

<fileset>.config A shell script that specifies configuration processing to be performed near the end of the installation process.

3.6.3 The Program Product Migration Process

The OPP and LPP migration process followed by the `installp` command is shown in flowchart form (of course!) in Figure 45 on page 155 and Figure 46 on page 156.

The process starts by restoring the ./lpp_name and liblpp.a files from the backup format file and extracting the files from the liblpp.a archive. It also checks the disk space requirements and fileset requesites using the information provided in the lpp_name file.

Next `installp` checks to see if the fileset is installed, either under the present name, or under one of the names listed in the <fileset>.namelist file. If not, this is just a simple fileset installation and no migration processing is necessary.

If a <fileset>.pre_rm script exists, it will now be executed.

Then, if <fileset>.cfgfiles exists, the files it lists will be saved to the directories:

- /usr/lpp/save.config for the usr part
- /lpp/save.config for the root part
- /usr/share/lpp/save.config for the share part

The files will be saved with the same relative path as their original location. These files will be processed later by the post_i routine.

Next, the files listed in <fileset>.inventory are removed from the system and from the SWVPD in the ODM to make way for their replacement files. Similarly, if a <fileset>.rm_inv file exists, the files it lists are also removed at this time.

The <fileset>.pre_i script, which is now run if it exists, will typically perform such actions as quiescing particular operations, or modifying or deleting System Resource Controller (SRC) subsystems.

The files associated with this fileset are now restored from the installation media and installed onto the system.

After installing the files the system checks for a <fileset>.post_i script. If the script exists, it will now be run. The post_i routine will most commonly be used to merge any configuration data or format changes from the new versions of configuration files into the old version stored in the save directories. If <fileset>.cfgfiles still
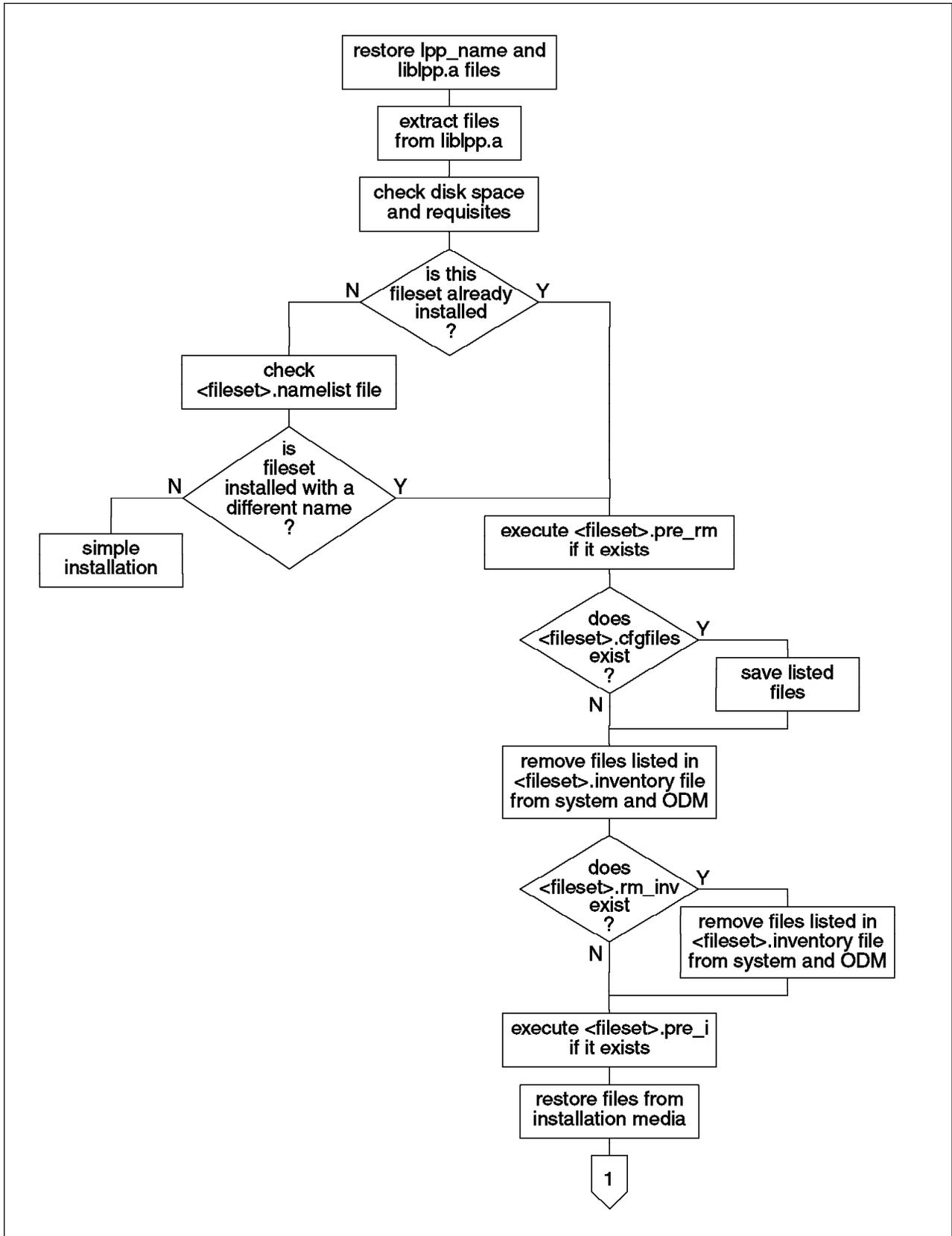
Figure 45. Program Product Migration Process - Part A.
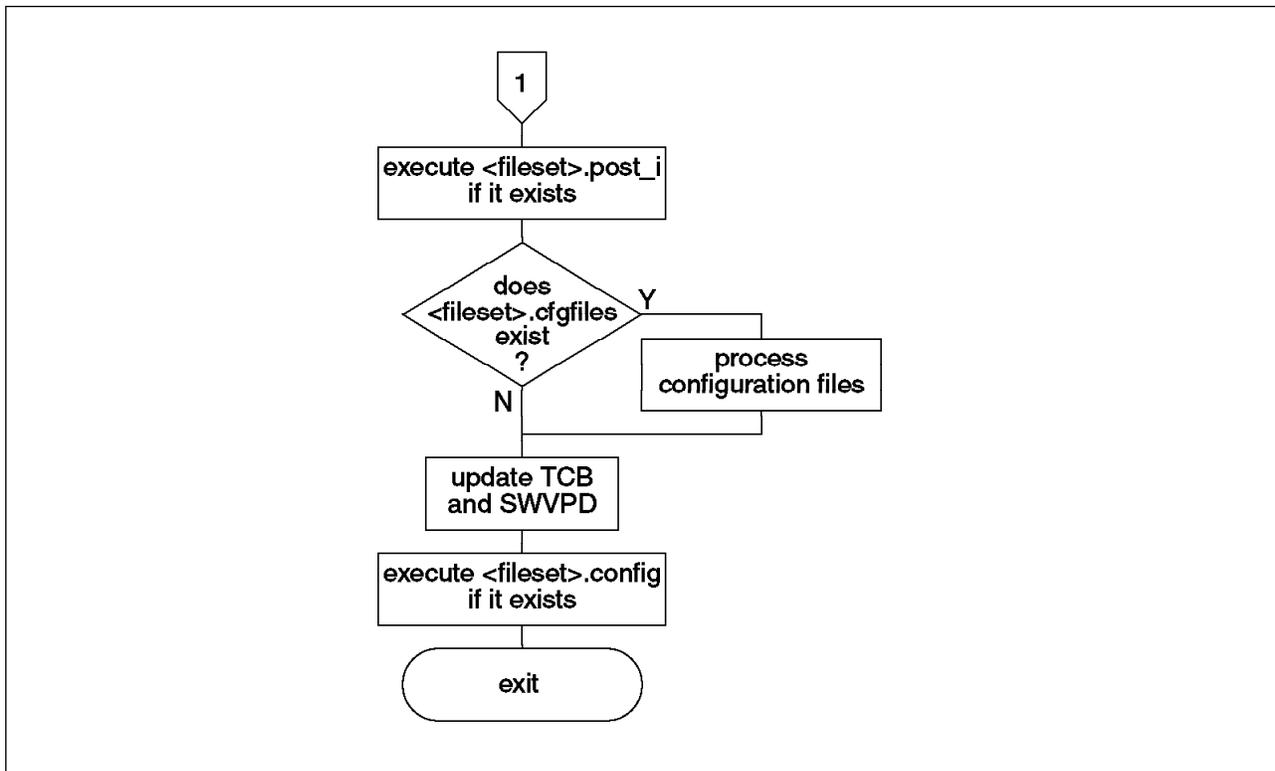
Figure 46. Program Product Migration Process - Part B.

exists, the listed files will then be returned to their original positions or left in the save directories as specified by their associated action keywords.

Next `sysck` is called to update the SWVPD with the newly installed files, and `tcbck` to update the Trusted Computing Base (TCB) if appropriate. Various other ODM updates are also performed at this time.

Finally, the system checks for, and executes if available, a script named <fileset>.config which will typically perform such actions as updating or creating SRC subsystems and other customizations.

3.7 Device Configuration

Device configuration has changed very little between AIX Version 3.2 and AIX Version 4.1. In both cases, the command `cfgmgr` detects and configures devices.

When `cfgmgr` is called, it examines the Object Database Manager (ODM) database Config_Rules. The Config_Rules object class defines rules for each device configuration method including:

- The full path of the configuration method to run

- The phase in which to run the method

- A relative sequence within the phase

- A mask field

Each configuration method contains the specific code required to detect and define a device or class of devices. In many cases the specific configuration methods will

set the LED display on the RISC System/6000 front panel to indicate the device being configured. Thus there are LED codes that will be seen during the boot process that are not shown on these flowcharts.

The device configuration process defines two phases of operation, although the phase parameter (-p) can be set to the values 1, 2 or 3, where phase=3 indicates the second phase of configuration during a service mode boot. These phases do not necessarily correlate with the phases of the boot process.

The relative sequence is an integer value that indicates the order in which the devices should be configured within a phase. In general, a lower number indicates a higher priority, with the exception that the value 0 indicates a "don't care" status and thus entries with a sequence value of zero are executed last.

The mask field is not often used but allows specific devices to be run, or to be omitted from a configuration phase.

One important factor related to device configuration when migrating an existing system is the fact that under certain conditions, the names allocated to devices can change. This problem is described below.

For more information on the `cfgmgr` command and the device configuration process, see the following sources:

- *InfoExplorer*
- *AIX Version 3.2 Commands Reference*
- *AIX Version 3.2 System Management Guide: Operating System and Devices*
- *AIX Version 4.1 Commands Reference*
- *AIX Version 4.1 System Management Guide: Operating System and Devices*

3.7.1 Device Naming Problems

On a running system when booted in normal mode, when `cfgmgr` discovers a device it will look for this device in the ODM databases to see if this device has been configured before. If it finds the device it will configure the device with the same name and any specific device customizations that were previously used. If the device is unknown, it will be assigned the next available name for a device of that type and default values for any configurable parameters.

However, if the system is booted in service mode either to perform diagnostics or to reinstall the system, the configured ODM databases are not available. In this case, all devices will be unknown and thus will be named in the order they are discovered. This may lead to changed device names, as shown in the following example:

Consider a system with three disks as shown below. The three disks have SCSI addresses 0, 1 and 2, and are connected to the standard SCSI adapter in slot 8 of the system.

```
+ hdisk0          00-08-00-0,0      857 MB SCSI Disk Drive
+ hdisk1          00-08-00-1,0      857 MB SCSI Disk Drive
+ hdisk2          00-08-00-2,0      857 MB SCSI Disk Drive
```

Later, the customer requires more disk, so they add three more disks. For performance reasons, they add a second SCSI adapter for the new disks. The new disks will be allocated the next available names, and thus will be called hdisk3, hdisk4 and hdisk5. The new disks also have addresses 0, 1 and 2 and are connected to the new adapter in slot 7.

```
+ hdisk0          00-08-00-0,0      857 MB SCSI Disk Drive
+ hdisk1          00-08-00-1,0      857 MB SCSI Disk Drive
+ hdisk2          00-08-00-2,0      857 MB SCSI Disk Drive
+ hdisk3          00-07-00-0,0      1.07 GB SCSI Disk Drive
+ hdisk4          00-07-00-1,0      1.07 GB SCSI Disk Drive
+ hdisk5          00-07-00-2,0      1.07 GB SCSI Disk Drive
```

When this system is booted from installation media to be reinstalled, the ODM database containing this information is not available. Thus the disks will be named in the order discovered. Since the SCSI adapter in slot 7 will be configured before the adapter in slot 8, the new disk names will be as shown below.

```
+ hdisk0          00-07-00-0,0      1.07 GB SCSI Disk Drive
+ hdisk1          00-07-00-1,0      1.07 GB SCSI Disk Drive
+ hdisk2          00-07-00-2,0      1.07 GB SCSI Disk Drive
+ hdisk3          00-08-00-0,0      857 MB SCSI Disk Drive
+ hdisk4          00-08-00-1,0      857 MB SCSI Disk Drive
+ hdisk5          00-08-00-2,0      857 MB SCSI Disk Drive
```

If care is not exercised, this could lead to unexpected results such as installing on the wrong disks and the loss of data. The same problem can occur with many other devices such as terminals. One case of this is described in Chapter 4, "UP to SMP Migration" on page 171 relating to the naming of terminal devices. This naming could also cause problems to systems running database software that uses raw disk devices. It is possible that configuration files and utilities such as backup scripts will refer to the /dev/hdiskx names in various places. The system or database administrators of these systems must be aware that the names may have changed.

The safest way to reliably identify devices is by their location code. Unless deliberate action is taken to change the address of an existing device, this will not change through the software installation process. Thus, as with purchasing real-estate, you should remember the golden rule—the three most important things are location, location and location.

3.8 Installation Customization Files

The most common and powerful method of customizing the installation process is by customizing the contents of the following files:

- bosinst.data

- image.data

- preserve.list

Together these files allow a great degree of control over the actions performed during the process of installation, the filesets that are installed, and the disks they

are installed onto. The contents of these files are described below. A user tailored copy of one or more of these files can be introduced into the installation process through one of two methods:

- Using the Network Installation Management (NIM)

  In the case of a NIM installation, these files can be defined as NIM resources and specified for each client system to be installed. For more information on this method, see *AIX Version 4.1 Network Installation Management Guide and Reference.*

- Placing the files in the root directory when performing a system backup

  This method only applies when installing the system from a system backup created with the `mksysb -i` command. The `mksysb -i` command will automatically create an image.data file. If a bosinst.data file is placed in the root directory (/) before creating the backup, this file will be restored and processed during the system installation.

- Using a supplementary diskette

  For installations from tape, CD-ROM or using NIM, a supplementary diskette can be used to provide these files. To create a supplementary diskette, you must first create an ASCII file named signature containing just the word "data". This can be done using a text editor such as `vi` or `emacs`, or by using a command such as:

  ```
  echo "data" >signature
  ```

  In the same directory, place your tailored versions of the required files. You can provide any combination of the above files, and could also provide files to describe user defined bundles of software to be installed. Then use the following command to backup the files to diskette:

  ```
  ls ./signature ./bosinst.data ./image.data ./mybundle | backup -ivq
  ```

  List each file that you wish to copy to the diskette. Ensure that you use the relative path for the files (preceed the files by ./). As is shown in Figure 10 on page 99, after booting from the installation media, the `bi_main` script will retrieve these files from the installation media before searching for the ./signature file on a diskette. If the ./signature file is found and contains the string "data", all files on the diskette will be restored and will thus overwrite the default files provided with the installation media, including any files placed in the root directory during a system backup.

Most of the information in this and the following sections comes from the *AIX Version 4.1 Installation Guide*, and from the *AIX Version 4.1 Files Reference*. See these volumes or *InfoExplorer* for further details.

At the completion of the installation the completed versions of these files with the values that were actually used for the installation will be placed in the directory /var/adm/ras for later reference.

3.8.1 bosinst.data file

The bosinst.data allows great control over the installation process followed and the software installed, allowing the user, for example, to:

- install the BOS without any user interaction
- specify the default values to be used in the installation menus
- list other filesets to be automatically installed
- specify target disks

A sample bosinst.data file can be found in the directory /var/adm/ras on an AIX Version 4.1 system. The file contains many variables arranged into stanzas. The meaning of these variables are described below:

- **control_flow** stanza

    CONSOLE The full path name of the device to be used as the console. If this field is blank you will be prompted to press a number or function key to select the console.

    INSTALL_METHOD Specifies the method of installation to be used.

    PROMPT Specifies if the installation menus should be used. If PROMPT is set to no, values must be supplied for all variables in the locale stanza and all variables in the control_flow stanza except ERROR_EXIT and CUSTOMIZATION_FILE. If one or more of the variables are found to contain illegal or contradictory values, or another error occurs, PROMPT will be set to yes and an error message will be displayed on the first menu screen.

    EXISTING_SYSTEM_OVERWRITE Permits the system to overwrite existing volume groups.

    INSTALL_X_IF_ADAPTER If this variable is set to **yes**, AIXWindows will be installed if the selected console is a graphics terminal. If the value is **no**, AIXWindows will not be installed. If the value **all** is specified, AIXWindows will be installed regardless of the console.

    RUN_STARTUP Specifies if the Installation Assistant program should be run on the first system boot after installation.

    RM_INST_ROOTS Specifies whether to remove the files and directories /usr/lpp/*/inst_roots. These files are required only if the system is to be used as a network installation server.

    ERROR_EXIT Allows the user to specify their own error-handling routine to be run in the case of an error. This routine could be a shell script or program and could be restored from the supplemental diskette along with the other files.

    CUSTOMIZATION_FILE This variable can be used to specify a customization script or program to run as the last step in the installation process. This customization script could also be restored from the supplemental diskette if required.

    TCB Specifies whether to install the Trusted Computing Base (TCB), which must be installed at system installation time. For more information on the TCB, see *InfoExplorer* or *AIX Version 4.1 System Management Guide: Operating System and Devices*, SC23-2525.

We strongly recommend not selecting yes for TCB installation at AIX levels below modification level 4.1.3.

INSTALL_TYPE Specifies the software to install on the system from a set of predefined configurations. The options are a client configuration, a personal workstation configuration, or a full configuration which includes all the contents of the client and personal selections.

BUNDLES Specifies the names of software bundles to install. These can be system defined bundles or user bundles defined by a bundle file restored from a supplemental diskette or saved in a system backup. For installation from NIM or CD-ROM, the system bundles should be specified as:

`/SPOT/usr/sys/inst.data/sys_bundles/BundleName`

For tape installations, the path:

`/usr/sys/inst.data/sys_bundles/BundleName`

If a bundle file is stored on the supplemental diskette with the name `./MyBundleName`, the BUNDLES value should be set to:

`/../MyBundleName`

- **target_disk_data** stanzas

This variable defines the disks to be used as the root volume group for the newly installed system. One stanza should be used for each disk.

LOCATION Defines the location of the disk in the format XX-XX-XX-XX (or XX-XX-XX-X,X for SCSI devices). For more information on device location codes, see *POWERstation and POWERserver Common Diagnostics and Service Guide* or *Common Diagnostics Information Manual*.

SIZE_MB Specifies the formatted size of the disk in megabytes. The command `bootinfo -s hdiskname` can be used to obtain the formatted size of a disk. The value **largest** can be used to specify the single largest disk not already selected.

HDISKNAME Specifies the name of the target disk.

> ┌─ **Attention** ─────────────────────────────────┐
>
> Note that the name of the disk may not match the name it used in a previous software installation on the same software. For a description of this problem see 3.7, "Device Configuration" on page 156. By far the safest and most accurate method of specifying a particular device is to use the location code.

The variables are evaluated in the priority order above. If a location code is specified then the disk at that location will be used, regardless of the size or name specified.

- **locale** stanza

This stanza specifies the language, keyboard and cultural conventions that the installed system will use. For more information on the values that can be used in this field, see *AIX Version 4.1 System User's Guide: Operating System and Devices*, SC23-2544. This stanza contains the variables:

BOSINST_LANG The language to be used during the installation process.

CULTURAL_CONVENTION The locale to install.

MESSAGES The message catalogs to install.

KEYBOARD The keyboard map to be used.

Table 10 shows the format or possible values for each variable and the default value.

Table 10. bosinst.data Variables		
Variable	**Format or Permitted Values**	**Default**
CONSOLE	Examples: /dev/ttyx, /dev/lft	blank (prompt user)
INSTALL_METHOD	migrate, preserve, overwrite	Varies—see Table 9 on page 103
PROMPT	yes, no	yes
EXISTING_SYSTEM _OVERWRITE	yes, no	no
INSTALL_X_IF_ADAPTER	yes, no, all	yes
RUN_STARTUP	yes, no	yes
RM_INST_ROOTS	yes, no	no
ERROR_EXIT	Example: ./my_error_proc	blank
CUSTOMIZATION_FILE	Example: /my_custom_stuff	blank
TCB	yes, no	no
INSTALL_TYPE	client, personal, full	Depends on software purchased
BUNDLES	/SPOT/usr/sys/inst.data/sys_bundles/BundleName (for CD-ROM or NIM) /usr/sys/inst.image/sys_bundles/BundleName (for tape) /../MyBundleName (for a user bundle)	
LOCATION	XX-XX-XX-XX, XX-XX-XX-X,X	blank
SIZE_MB	Examples: 382, 305, 817, 1022, largest	blank
HDISKNAME	Examples: hdisk0, hdisk1	blank
BOSINST_LANG	Examples: en_US, da_DK	blank
CULTURAL_CONVENTION	Examples: en_US, da_DK	blank
MESSAGES	Examples: en_US, da_DK	blank
KEYBOARD	Examples: en_US, da_DK	blank

A sample bosinst.data file is shown below.

```
control_flow:
 CONSOLE = /dev/lft0
 INSTALL_METHOD = migrate
 PROMPT = no
 EXISTING_SYSTEM_OVERWRITE = yes
 INSTALL_X_IF_ADAPTER = yes
 RUN_STARTUP = no
 RM_INST_ROOTS = no
 ERROR_EXIT =
 CUSTOMIZATION_FILE = /../my_custom
 TCB = no
 INSTALL_TYPE = personal
 BUNDLES =
target_disk_data:
 LOCATION =
 SIZE_MB =
 HDISKNAME =
locale
 BOSINST_LANG = en_US
 CULTURAL_CONVENTION = en_US
 MESSAGES = en_US
 KEYBOARD = en_US
```

3.8.2 image.data file

The image.data file uses a similar stanza-based format to the bosinst.data file, and in fact some of the same information is duplicated between the two files. In general, however, the image.data file is more oriented towards the volume groups, logical volumes and file systems that were on the system when it was backed up.

The stanzas appearing in the image.data file and a description of some of the variables are listed below:

- image_data

  Includes variables for the type of backup image, the date and time the backup was taken, a list of user volume groups on the system and an indication whether the tape is a product distribution tape or a mksysb backup.

- logical_volume_policy

  Sets overall policy for the creation of logical volumes including whether the logical volumes should be created the same size as before or shrunk to the minimum size required to contain the data on the file systems.

- ils_data

  Specifies the language to use for the installation programs.

- source_disk_data

  The image.data file contains one source_disk_data stanza for each disk in the root volume group. For each disk, the stanza lists the disk name, location and size in megabytes.

- vg_data

  This stanza specifies the name, physical partition size and the list of disks that contain the root volume group.

- lv_data

  The image.data file includes one lv_data stanza for each logical volume in the root volume group. This stanza includes all of the logical volume parameters

including the number of copies, allocation policies, type, size and optionally the path of a map file to use when recreating the logical volume.

- fs_data

  Again, the file contains one stanza per file system listing the file system size, mount point, fragmentation and compression. It also lists the minimum size that the file system could be shrunk to while still able to hold all of the data.

- post_install_data

  Allows the user to specify a command to be run after the completion of a BOS install. This stanza will only appear in rootvg (system) backups.

- post_restvg

  Specifies a command to run after the completion of a `restvg` command. This stanza will only appear in non-rootvg backups.

The image.data file is intended to capture the configuration of a system when making a system backup using the `mksysb` command to make a system backup. It replaces the .fs.size file that was used under AIX Version 3.2.

The image.data file is created automatically by the `mkszfile` command on an AIX Version 4.1 system. Great care must be taken when editing this file as there are many dependencies and links between the different stanzas and it is easy to finish with a corrupted file. The most commonly modified is the SHRINK variable in the image_data stanza. This field can be set to `yes` to indicate that the logical volumes and file systems should be created at the minimum size required to contain their data.

A sample image.data file is shown below.

```
image_data:
    IMAGE_TYPE= bff
    DATE_TIME= Fri Jan 12 16:29:45 CST 1996
    UNAME_INFO= AIX ramjet 1 4 000033751800
    PRODUCT_TAPE= no
    USERVG_LIST= camvg

logical_volume_policy:
    SHRINK= yes
    EXACT_FIT= no

ils_data:
    LANG=   en_US

##Command used for vg_data; /usr/sbin/lsvg

vg_data:
    VGNAME= rootvg
    PPSIZE= 4
    VARYON= yes
    VG_SOURCE_DISK_LIST= hdisk0 hdisk1

##Command used for source_disk_data; /usr/sbin/bootinfo

source_disk_data:
    LOCATION= 00-08-00-0,0
    SIZE_MB= 382
    HDISKNAME= hdisk0

source_disk_data:
    LOCATION= 00-08-00-1,0
    SIZE_MB= 382
    HDISKNAME= hdisk1

##Command used for lv_data; /usr/sbin/lslv

lv_data:
    VOLUME_GROUP= rootvg
    LV_SOURCE_DISK_LIST= hdisk0
    LV_IDENTIFIER= 000033753ce2e64a.1
    LOGICAL_VOLUME= hd6
    VG_STAT= active/complete
    TYPE= paging
    MAX_LPS= 128
    COPIES= 1
    LPs= 20
    STALE_PPs= 0
    INTER_POLICY= minimum
    INTRA_POLICY= middle
    MOUNT_POINT=
    MIRROR_WRITE_CONSISTENCY= off
    LV_SEPARATE_PV= yes
    PERMISSION= read/write
    LV_STATE= opened/syncd
    WRITE_VERIFY= off
    PP_SIZE= 4
    SCHED_POLICY= parallel
    PP= 20
    BB_POLICY= non-relocatable
    RELOCATABLE= yes
    UPPER_BOUND= 32
    LABEL=
    MAPFILE=
    LV_MIN_LPS= 20
```

```
lv_data:
    VOLUME_GROUP= rootvg
    LV_SOURCE_DISK_LIST= hdisk0
    LV_IDENTIFIER= 000033753ce2e64a.5
    LOGICAL_VOLUME= hd4
    VG_STAT= active/complete
    TYPE= jfs
    MAX_LPS= 128
    COPIES= 1
    LPs= 2
    STALE_PPs= 0
    INTER_POLICY= minimum
    INTRA_POLICY= center
    MOUNT_POINT= /
    MIRROR_WRITE_CONSISTENCY= on
    LV_SEPARATE_PV= yes
    PERMISSION= read/write
    LV_STATE= opened/syncd
    WRITE_VERIFY= off
    PP_SIZE= 4
    SCHED_POLICY= parallel
    PP= 2
    BB_POLICY= relocatable
    RELOCATABLE= yes
    UPPER_BOUND= 32
    LABEL= /
    MAPFILE=
    LV_MIN_LPS= 2

Note - Other lv_data stanzas deleted

##Commands used for fs_data; /usr/bin/df and /usr/sbin/lsjfs

fs_data:
    FS_NAME= /
    FS_SIZE= 16384
    FS_MIN_SIZE= 10216
    FS_LV= /dev/hd4
    FS_FS= 4096
    FS_NBPI= 2048
    FS_COMPRESS= no

Note - Other fs_data stanzas deleted

post_install_data:
    BOSINST_FILE=
```

3.8.3 preserve.list file

The preserve.list file is a simple list of file names—one name per line. The file can include comments by starting comment lines with the # character.

The preserve.list file is only used during a preservation installation. The files listed are copied to the /tmp/bos directory before the system file systems are deleted and recreated. The files are later copied back to their original positions.

The default preserve.list file is shown below.

```
# @(#) 43 1.1 src/bos/usr/lpp/bosinst/preserve.list/preserve.list, bosinst,
bos411, 9428A410j 6/30/93 17:56:40
#
#   COMPONENT_NAME: BOSINST
#
#   FUNCTIONS: none
#
#   ORIGINS: 27
#
#
#   (C) COPYRIGHT International Business Machines Corp. 1993
#   All Rights Reserved
#   Licensed Materials - Property of IBM
#   US Government Users Restricted Rights - Use, duplication or
#   disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#
#
# This file contains a list of system files which will be copied during a
# preservation BOS install.  Any configuration files which the user wishes to
# save during a preservation BOS install should be added to the following list.
# The file name must contain the full directory path.

/etc/filesystems
```

Part 2. Uniprocessor to Symmetric Multiprocessor

Chapter 4. UP to SMP Migration

This chapter covers the migration of a uniprocessor (UP) system running AIX V3.2.5 to an SMP system. Its objective is to help you to prepare and perform a UP to SMP upgrade.

A migration from a uniprocessor to a multiprocessor is not just a case of loading a new operating system and testing the environment. The SMP technology is very different from the UP technology, and going from one to the other must be done carefully.

Because of the changes that have occurred between the two architectures, there are quite a few questions that have to be answered regarding the system environment before starting any migration to an SMP system.

- Has my application been certified to work with AIX Version 4.1?
- If my application has been verified to work with AIX Version 4.1, is it SMP safe?
- Are my existing licensed products able to work with AIX Version 4.1?
- Must I have new license keys generated for my software?
- How does one change the kernel to take advantage of an SMP?
- Can I transfer the existing microchannel adapters and associated hardware to the SMP system?
- If the adapters and hardware cannot be transferred, what is available to take their place?
- Is there any third-party hardware installed in the system?
- If there is third-party hardware installed, is the hardware supported on the SMP system running AIX V4.1, and do I need additional device drivers?

4.1 Related Publications

The following publications may be useful during the migration of a uniprocessor to a multiprocessor.

- *IBM RISC System/6000 SMP Servers Architecture and Implementation*, SG24-2583
- *7012 G Series Operator Guide*, SA23-2740
- *7012 G Series Service Guide*, SA23-2741
- *7013 J Series Operator Guide*, SA23-2724
- *7013 J Series Service Guide*, SA23-2725
- *7015 Model R30 CPU Enclosure Operator Guide*, SA23-2742
- *7015 Model R30 CPU Enclosure Service Guide*, SA23-2743
- *7015 Model R00 Rack Installation and Service Guide R00*, SA23-2744

See also "Related Publications" on page xxi for publications related to AIX.

4.2 Model Conversions

Below, you will find the uniprocessor models that can be upgraded to the SMP models.

4.2.1 UP to G30

The following models can be upgraded to the G30:

- 7012 model 340
- 7012 model 34H
- 7012 model 350
- 7012 model 360
- 7012 model 370
- 7012 model 380
- 7012 model 390

4.2.2 UP to J30

The following models can be upgraded to the J30:

- 7013 model 520
- 7013 model 52H
- 7013 model 530
- 7013 model 53E
- 7013 model 53H
- 7013 model 540
- 7013 model 550
- 7013 model 55E
- 7013 model 55L
- 7013 model 55S
- 7013 model 560
- 7013 model 56F
- 7013 model 570
- 7013 model 57F
- 7013 model 580
- 7013 model 58H
- 7013 model 58F
- 7013 model 590
- 7013 model 59H

4.2.3 UP to R30

The following models can be upgraded to the R3U:

- 7015 model 930
- 7015 model 950
- 7015 model 950E
- 7015 model 970
- 7015 model 970B
- 7015 model 970E
- 7015 model 970F
- 7015 model 980
- 7015 model 980B
- 7015 model 980E
- 7015 model 980F
- 7015 model 990

- 7015 model 990E
- 7015 model 990F
- 7015 model 99J
- 7015 model 99K

The following models can be upgraded to the R30:

- 7015 model R10

- 7015 model R20

- 7015 model R24

4.3 Planning the Upgrade

In order to help you in preparing your SMP upgrade, you will find below a list of items that should be checked. This is not an exhaustive list and should be used as a guideline. The rest of this document assumes that your uniprocessor system is running AIX V3.2.5.

- Third-party applications:

  Check with the vendor of each third-party application that the application has been certified for AIX Version 4.1 and is SMP safe. If the application has not been certified for AIX Version 4.1, the migration cannot be performed. Also, if the application has been certified for AIX Version 4.1 but is not SMP safe, the migration cannot be performed.

- Custom-made applications:

  Check for any custom-made applications. These custom-made applications might have to be ported to AIX Version 4.1 in order for them to run on an SMP system. If the custom-made applications have not, or cannot, be ported to AIX Version 4.1, the migration cannot be performed.

- LPP migration paths:

  Check the installed LPPs, and verify that they are available in AIX V4.1 If yes, check the actual upgrade path to the AIX V4.1 version of each LPP (same or new product, free or fee upgrade). Check also to see if the AIX V4.1 equivalent LPP is SMP safe. Refer to *A Holistic Approach to AIX V4.1, Planning Guide*. If some required LPPs are not available in AIX V4.1 or not SMP safe, the migration cannot be performed.

- License keys:

  Check for any products that have an iFOR/LS license key. The CPU-ID will not be kept through the upgrade to the SMP; thus new keys for installed products requiring a license key will be needed. Check also to see if installed products that do not require license keys in AIX V3.2.5 will require license keys in AIX V4.1.

- IBM hardware compatibility:

  Check all installed adapters and I/O devices for compatibility with the SMP system. If the installed adapters and I/O devices are not compatible, have replacement adapters and I/O devices been ordered? If the replacement adapters and I/O devices have not been ordered and delivered, the migration cannot be performed. Check at the end of this chapter to see which IBM hardware is not supported on the SMP family of systems.

If the installed adapters and I/O devices are not compatible with the SMP
system, and there are no substitute adapters and I/O devices, are you aware of
this? Do you have an alternative to the non-compatible adapters and I/O
devices? If you do not have any alternatives to the non-compatible adapters
and I/O devices, then the migration cannot be performed.

- Third-party hardware compatibility:

The list of third-party hardware vendors that manufacture hardware for the IBM
RISC System/6000 is quite large; therefore IBM has no way of verifying what
third-party hardware will work with new versions of AIX.

If you have any third-party hardware installed in your system, IBM recommends
that you contact the hardware manufacturer, and ask the manufacturer if his
product has been tested with the version of AIX that you are installing.

If the hardware has been tested and certified with the version of AIX that you
are installing, check if there are any additional device drivers for the hardware
or if a generic device driver that is standard with AIX can be used.

Customers or IBM employees that require information regarding the description
of the hardware and/or physical addresses and phone numbers of third-party
hardware vendors can obtain the information at the following locations:

External WWW site:

`http://www.austin.ibm.com/developer/aix/library/oemhw/oemhw_2.html`

Internal IBM WWW site:

`http://info6000.austin.ibm.com/developer/aix/library/oemhw/oemhw_2.html`

If the third-party hardware is not supported by a multiprocessor system, the
hardware will not be migrated to the multiprocessor system, and the migration
cannot be performed until support is available or until the customer purchases
supported hardware.

4.3.1 AIX V4.1 Levels for SMPs

At the time this document was written, three modification levels of AIX V4.1
supported the SMP systems.

- AIX V4.1.2
- AIX V4.1.3
- AIX V4.1.4

AIX V4.1.2 supports SMP configuration up to four processors. AIX V4.1.3 supports
up to eight processors and has been optimized for eight processors.

We recommend that you upgrade your uniprocessor system directly to AIX V4.1.3
or AIX V4.1.4, unless some software is only supported in AIX V4.1.2.

Note: We tested the migration from a UP to SMP from an AIX V3.2.5 UP to an
AIX V4.1.3 SMP.

4.3.2 Platform Types

When dealing with both UPs and SMPs, it is important to understand the concept of system device and hardware platform type.

First of all, the system device, *sys0*, is the highest-level device in the system. It consists of all physical devices in the system.

The hardware platform type is an abstraction that allows machines to be grouped according to fundamental configuration characteristics, such as number of processors and/or I/O bus structure. Machines with different platform types have basic differences in the way their devices are dynamically configured at boot time.

rs6k   applies to all of the uniprocessor IBM RISC System/6000 that have a microchannel bus.

rs6ksmp applies to symmetric multiprocessor models.

rspc   applies to the PReP-compliant systems that have an ISA bus and PCI bus, such as the RISC System/6000 40P and 43P.

Prototype files are used by the `bosboot` command to build the boot image for a specific boot device. In AIX V3.2.5, prototype files are dependent on the boot device. In AIX V4.1, the prototype files are dependent on the boot device AND on the hardware platform type of the machine for which the boot image is built.

This is reflected in the names of the prototype files:

/usr/lib/boot/rs6k.cd.proto
/usr/lib/boot/rs6k.disk.proto
/usr/lib/boot/rs6k.tape.proto
/usr/lib/boot/rs6ksmp.cd.proto
/usr/lib/boot/rs6ksmp.disk.proto
/usr/lib/boot/rs6ksmp.tape.proto
/usr/lib/boot/rspc.disk.proto
/usr/lib/boot/rspc.cd.proto
/usr/lib/boot/network/rs6k.ent.fddi
/usr/lib/boot/network/rs6k.fddi.proto
/usr/lib/boot/network/rs6k.tok.proto
/usr/lib/boot/network/rs6ksmp.ent.proto
/usr/lib/boot/network/rs6ksmp.tok.proto
/usr/lib/boot/network/rspc.ent.proto
/usr/lib/boot/network/rspc.tok.proto

These files, in addition to the configuration methods, are contained in the following filesets:

devices.base.*

devices.rs6ksmp.base.*

devices.rspc.base.*

The `bootinfo` command is used during the boot phases to gather and display information on the platform type, current boot device and so on. This command has a set of options that allows the system to retrieve needed information at boot time.

These are some of the `bootinfo` command options:

`bootinfo -z` will return:

0   if the machine is not MP-capable
1   if the machine is MP-capable

`bootinfo -T` will return one of the following platform types:

rs6k   which means RISC System/6000 UP
rs6ksmp   which means RISC System/6000 SMP
rspc   which means PowerPC PReP system

`bootinfo -t` will list the type of boot, and the following are the responses:

1   Disk boot
3   CD-ROM boot
4   Tape boot
5   Network boot

`bootinfo -k` will display the key position:

1   Secure position
2   Service position
3   Normal position

4.3.3 UP and MP Kernels

AIX V4.1 is shipped with two kernels - an MP kernel and a UP kernel. The two kernels have common code, but the locking required for an SMP environment is eliminated for the UP kernel. There are three key filesets in this regard.

bos.rte.*   These filesets are used to ship BOS runtime files that are common to both the UP and the MP systems.

bos.rte.up This fileset contains files specific to a UP (like the UP kernel).

bos.rte.mp This fileset contains files specific to an SMP (like the MP kernel)

Note:

When installing a system, the correct kernel will be installed, depending on the type of system you are dealing with. In fact, as part of the BOS installation, the command `bootinfo -z` determines whether the system is MP capable or not. Then the BOS installation process installs the appropriate fileset and link /unix to the correct kernel. This link is only used by the `bosboot` command when rebuilding the boot image.

The bos.rte.mp fileset contains the following files:

```
# lslpp -f bos.rte.mp
        Fileset                 File
  --------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.rte.mp 4.1.4.0     /usr/sbin/open_door
                         /usr/sbin/cpu_state
                         /usr/sbin/bindprocessor
                         /usr/lib/boot/unix_mp

Path: /etc/objrepos
  bos.rte.mp 4.1.4.0     NONE
```

The file /usr/lib/boot/unix_mp is the MP kernel. The /usr/sbin/cpu_state and /usr/sbin/bindprocessor files are SMP specific. For more information on these files, refer to *IBM RISC System/6000 SMP Servers Architecture and Implementation*.

Note: It is possible to install both the UP and the MP filesets on the same system, but see the following section for boot images issues.

4.3.4 MP Boot Image

In AIX V4.1, the `bosboot` command is used to create a boot image, and this is the same as in AIX V3.2. This command uses the prototype files listed in the Platform Types section.

However, any boot image will only support a single platform type and a single boot device type. The supported boot devices are token-ring, Ethernet, FDDI, CD-ROM, disk, or tape.

The command you would use to create an MP boot image is:

```
# bosboot -a -d hdiskx -k /usr/lib/boot/unix_mp -T <platform_type>
```

where

- `-a` Creates a complete boot image and device.
- `-d` Specifies the boot device. This flag is optional for the hard disk.
- `-k` Specifies the full path to the MP kernel. This is optional if /unix is already linked to /usr/lib/boot/unix_mp.
- `-T` Specifies the platform type for the boot image. This is optional since the platform type is taken from the system where the `bosboot` command is run. But if you want to create on a UP system an MP boot image for an SMP system, you need to specify the type, `rs6ksmp`.

> ⎯ **Attention!** ⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯⎯
>
> A **UP kernel will not work on an MP system** because there is some MP hardware-specific support that is not included in the UP kernel. You will get an LED code of 911 if you try to boot an SMP system with a UP boot image.
>
> An **MP kernel will work on a UP system** if, when creating the boot image, you specified the corresponding UP platform type or used the default. If you specified the platform type rs6ksmp when creating the MP boot image, you will not be able to boot on the UP.

4.3.5 CPU ID

On UP systems, the output of the `uname -m`, which gives the CPU ID, is tied to the CPU card.

On an SMP, the CPU ID is not tied to a processor card. The CPU ID is part of the VPD (Vital Product Data) of the system and is stored in the System EEPROM. The `uname` command will use the information stored there to build the CPU ID.

The CPU ID cannot be maintained through a UP to SMP upgrade. This means that it will be necessary to ask for new keys for licensed products. For more information on the SMP CPU ID, refer to *IBM RISC System/6000 SMP Servers Architecture and Implementation*, SG24-2583.

4.3.6 Terminal and Printer Migration Issues

When a system is migrated from a uniprocessor to a multiprocessor, the tty and printer configurations will have to be recreated. This is because the system numbering (for example; sa0) of the asynchronous ports and adapters changes. The uniprocessor has two (2) serial ports, and the multiprocessor has three (3) serial ports, by default.

```
Uniprocessor

sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2

Multiprocessor

sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2
sa2        Available 00-00-S3     Standard I/O Serial Port 3
```

If the system has an 128-port adapter installed, the system numbering on a uniprocessor for the first 16-port Remote Asynchronous Node (RAN) EIA-232 for the 128-port adapter will be sa2, and on a multiprocessor, the system numbering will be sa3.

```
Uniprocessor

sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2
cxma0      Available 00-01        128-Port Asynchronous Adapter
sa2        Available 00-01-11     16-Port RAN EIA-232 for 128-Port Adapter

Multiprocessor

sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2
sa2        Available 00-00-S3     Standard I/O Serial Port 3
cxma0      Available 00-01        128-Port Asynchronous Adapter
sa3        Available 00-01-11     16-Port RAN EIA-232 for 128-Port Adapter
```

Hence, when the mksysb from the uniprocessor tries to restore the ttys and printers onto sa2 on the multiprocessor (trying to duplicate the uniprocessors configuration), the configuration methods routine will fail when the system tries to rebuild the Object Data Manager (ODM). This is because sa2 on the uniprocessor is the 16-port RAN for the 128-port adapter, and sa2 on the multiprocessor is the standard I/O serial port, 3.

All the ttys and printers that should be configured onto the 16-port RAN for the 128-port asynchronous adapter will be deleted and will not be configured onto the 16-port RAN for the 128-port asynchronous adapter.

When the multiprocessor is rebooted, none of the ttys or printers that should have been configured for the 128-port adapter will be available; they will have to be reconfigured.

After the system has been restored by a mksysb, there will be two files in /tmp that will enable you to recreate the printers and ttys. These files are:

- /tmp/reconfig1
- /tmp/reconfig2

A sample of the /tmp/reconfig1 file that will be created on the multiprocessor is listed here for your convenience.

```
#!/bin/ksh
export TALLY=
cmd_1='/etc/methods/define -l tty1 -c tty -s rs232 -t tty -p sa2 -w 0 '
cmd_2='/etc/methods/cfgtty -l tty1 '
cmd_3='/etc/methods/define -l tty2 -c tty -s rs232 -t tty -p sa2 -w 1 '
cmd_4='/etc/methods/cfgtty -l tty2 '
cmd_5='/etc/methods/define -l tty3 -c tty -s rs232 -t tty -p sa2 -w 2 '
cmd_6='/etc/methods/cfgtty -l tty3 '
cmd_7='/etc/methods/define -l tty4 -c tty -s rs232 -t tty -p sa2 -w 3 '
cmd_8='/etc/methods/cfgtty -l tty4 '
cmd_9='/etc/methods/define -l tty5 -c tty -s rs232 -t tty -p sa2 -w 4 '
cmd_10='/etc/methods/cfgtty -l tty5 '
cmd_11='/etc/methods/define -l tty6 -c tty -s rs232 -t tty -p sa2 -w 5 '
cmd_12='/etc/methods/cfgtty -l tty6 '
cmd_13='/etc/methods/define -l tty7 -c tty -s rs232 -t tty -p sa2 -w 6 '
cmd_14='/etc/methods/cfgtty -l tty7 '
cmd_15='/etc/methods/define -l tty8 -c tty -s rs232 -t tty -p sa2 -w 7 '
cmd_16='/etc/methods/cfgtty -l tty8 '
cmd_17='/etc/methods/define -l tty9 -c tty -s rs232 -t tty -p sa2 -w 8 '
cmd_18='/etc/methods/cfgtty -l tty9 '
cmd_19='/etc/methods/define -l tty10 -c tty -s rs232 -t tty -p sa2 -w 9 '
cmd_20='/etc/methods/cfgtty -l tty10 '
cmd_21='/etc/methods/define -l tty11 -c tty -s rs232 -t tty -p sa2 -w 10 '
cmd_22='/etc/methods/cfgtty -l tty11 '
cmd_23='/etc/methods/define -l tty12 -c tty -s rs232 -t tty -p sa2 -w 11 '
cmd_24='/etc/methods/cfgtty -l tty12 '
cmd_25='/etc/methods/define -l tty13 -c tty -s rs232 -t tty -p sa2 -w 12 '
cmd_26='/etc/methods/cfgtty -l tty13 '
cmd_27='/etc/methods/define -l tty14 -c tty -s rs232 -t tty -p sa2 -w 13 '
cmd_27='/etc/methods/cfgtty -l tty14 '
cmd_29='/etc/methods/define -l tty15 -c tty -s rs232 -t tty -p sa2 -w 14 '
cmd_30='/etc/methods/cfgtty -l tty15 '
cmd_31='/etc/methods/define -l tty16 -c tty -s rs232 -t tty -p sa2 -w 15 '
cmd_32='/etc/methods/cfgtty -l tty16 '
cmd_33='chdev -l tty0
        -a login=enable
        -a term=ibm3151'
cmd_34='chdev -l tty1
        -a login=enable
        -a term=ibm3151'
cmd_35='chdev -l tty2
        -a login=enable
        -a term=ibm3151'
cmd_36='chdev -l tty3
        -a login=enable
        -a term=ibm3151'
cmd_37='chdev -l tty4
        -a login=enable
        -a term=ibm3151'
cmd_38='chdev -l tty5
        -a login=enable
```

```
                        -a term=ibm3151'
        cmd_39='chdev -l tty6
                -a login=enable
                -a term=ibm3151'
        cmd_40='chdev -l tty7
                -a login=enable
                -a term=ibm3151'
        cmd_41='chdev -l tty8
                -a login=enable
                -a term=ibm3151'
        cmd_42='chdev -l tty9
                -a login=enable
                -a term=ibm3151'
        cmd_43='chdev -l tty10
                -a login=enable
                -a term=ibm3151'
        cmd_44='chdev -l tty11
                -a login=enable
                -a term=ibm3151'
        cmd_45='chdev -l tty12
                -a login=enable
                -a term=ibm3151'
        cmd_46='chdev -l tty13
                -a login=enable
                -a term=ibm3151'
        cmd_47='chdev -l tty14
                -a login=enable
                -a term=ibm3151'
        cmd_48='chdev -l tty15
                -a login=enable
                -a term=ibm3151'
        cmd_49='chdev -l tty16
                -a login=enable
                -a term=ibm3151'
        num_cmds=50
        i=0
        while [ $i -lt $num_cmds ]
        do
            eval \$cmd_$i > /dev/null
            if [ $? -ne 0 ]
            then
                TALLY="$TALLY$(eval echo \$cmd_$i)\n"
            fi
            (( i+=1 ))
        done
        if [ -n "$TALLY" ]
        then
            echo Machine not identical to previous configuration.
            echo The following items failed:
            echo $TALLY
        fi
```

We will explain in the migration procedure how to use these files to reconfigure the ttys and printers.

4.4 Migrating an AIX V3.2.5/UP to an AIX V4.1/SMP

When a uniprocessor is to be migrated to an SMP from AIX Version 3.2, not only is the operating system to be migrated but also the hardware. AIX has to be upgraded to AIX Version 4.1 first before any hardware changes can take place.

After the uniprocessor has been upgraded to AIX Version 4.1, the environment must be tested to be certain that the applications and AIX Version 4.1 are in order before the migration can continue.

After the environment has been tested and you are satisfied that the site is functional on AIX Version 4.1, you must install the filesets and device drivers to enable the multiprocessor kernel and the new hardware to be utilized.

After the filesets and device drivers have been installed and the multiprocessor kernel enabled, you are ready to migrate to a multiprocessor system. You will be able to back up the system, move the reusable hardware to the SMP and restore the system backup.

4.4.1 Migration Steps

The main steps in the migration from a UP system to an SMP system are the following:

- Check that all the necessary resources you need during the upgrade are available prior to the upgrade.
- Document the configuration of your system.
- Back up your entire UP system, including rootvg, user-defined volume groups and raw devices, if any.
- Migrate your UP system to AIX V4.1.
- Test your overall installation in AIX V4.1.
- Back up your AIX V4.1 UP installation.
- Install the MP kernel fileset that is required to enable the MP kernel.
- Install device drivers filesets to support SMP-specific devices.
- Create MP kernel links.
- Create an MP boot image.
- Remove device entries in the ODM.
- Create a bootable backup tape.
- Migrate the reusable hardware.
- Restore the system backup on your SMP system.
- Restore user-defined volume groups and raw devices
- Reconfigure asynchronous devices.
- Test the overall SMP installation.
- Back up the system.

4.4.2 Checking Resources

Make sure that you have the following resources available to you before proceeding with the upgrade. If these resources are not available, you will not be able to complete the upgrade.

The resources are:

- AIX V4.1 installation media (tape cartridge or CD-ROM).

- Enough tape cartridges to complete three system backups.

- An ASCII terminal with its keyboard.

- A half-high internal tape drive or an external tape drive or a CD-ROM drive.

- The hardware upgrade kit (checked for completeness) that should have been delivered with your SMP system.

- A list of the additional hardware, if any, that has been ordered with the upgrade.

  Note: Check that these additional features have been included in the upgrade kit. Depending on the procedures of the manufacturing plant that shipped this upgrade kit, some or all of these additional features may have been already installed within this upgrade. If they were factory installed, then a shipping list describing which were installed should have been included with the upgrade kit.

- Publications relating to the upgrade.

You should also be aware that any currently installed features that are not supported on the SMP system will not be moved to the upgraded machine.

Note that:

- The date and time will need to be reset upon completion of this upgrade.

- The system will be down for approximately six (6) to eight (8) hours.

4.4.3 Documenting the UP System

The following commands must be executed to determine the configuration of the uniprocessor.

Get information regarding the boot logical volume (hd5) by executing:

lslv -l hd5

The output of the command will be:

```
# lslv -l hd5
hd5:/blv
PV              COPIES          IN BAND      DISTRIBUTION
hdisk1          002:000:000     0%           000:002:000:000:000
#
```

Record the hdisk number that hd5 is stored on. This hdisk is the physical volume that has the boot logical volume (hd5) installed. You will need this information

when you execute the bosboot command to enable the multiprocessor kernel for the SMP system.

Get information about the physical volumes that are installed in the system by executing:

lspv

The output of the command will be:

```
# lspv
hdisk0         00004046cd7fc68c    rootvg
hdisk1         00000615bfbe9cd0    rootvg
#
```

Record the hdisk number(s) and the location of the physical volume(s) that the rootvg is stored on. You will need this information to bring up the system after the model upgrade is installed.

Get information regarding the VPD (Vital Product Data) of the physical volumes by executing:

lscfg -vl hdisk* | pg

The output of the command will be:

```
# lscfg -vl hdisk* | pg
  DEVICE            LOCATION          DESCRIPTION

  hdisk0            00-08-00-0,0      1.0 GB SCSI Disk Drive

    Manufacturer................IBM
    Machine Type and Model......0663L12
    Part Number.................45G9464
    ROS Level and ID............31303132
    Serial Number...............00026793
    EC Level....................895972
    FRU Number..................45G9467
    Device Specific.(Z0)........000002029F00001E
    Device Specific.(Z1)........45G9874
    Device Specific.(Z2)........09RI
    Device Specific.(Z3)........95188
    Device Specific.(Z4)........0002
    Device Specific.(Z5)........22
    Device Specific.(Z6)........895964

  hdisk1            00-08-00-1,0      1.0 GB SCSI Disk Drive

    Manufacturer................IBM
    Machine Type and Model......0663L12
    Part Number.................45G9464
:
```

Record the displayed Machine Type and Model and Serial Number fields for the hdisk(s) that contains the rootvg in Table 11 on page 184. Press the Enter (or

Return) key as needed to page down through the hdisk information, and return to the command prompt.

Table 11. rootvg Hard Drive Identification

Hdisk Number	Machine Type and Model	Serial Number
hdisk___		
hdisk___		
hdisk___		
hdisk___		
hdisk___		
hdisk___		
hdisk___		
hdisk___		

Get information regarding the devices that attach to the SCSI controllers by executing:

```
# lsdev -Cs scsi | pg
```

The output of the command will be:

```
# lsdev -Cs scsi | pg
hdisk0 Available 00-08-00-0,0 1.0 GB SCSI Disk Drive
hdisk1 Available 00-08-00-1,0 1.0 GB SCSI Disk Drive
rmt0   Available 00-08-00-5,0 2.3 GB 8mm Tape Drive
cd0    Available 00-08-00-6,0 CD-ROM Drive
(EOF):
```

Record the information displayed for each device attached to the SCSI adapters (for example, cd0), the device's SCSI address and (for disk drives only) the SCSI device's storage capacity, if given.

Read the section titled *Location Codes* in the *Diagnostic Numbers and Codes* chapter of the *Common Diagnostics and Service Guide* if you are not familiar with reading SCSI device address or SCSI adapter slot-number information from the displayed location code.

Table 12 (Page 1 of 2). SCSI Device Information

Slot Number	SCSI Type (00, 01 or 0S)	Device Name (rmt0)	SCSI Address (0 thru 6)	Hdisk Size
				GBytes
				GBytes
				GBytes
				GBytes
				GBytes
				GBytes
				GBytes

Table 12 (Page 2 of 2). SCSI Device Information				
Slot Number	**SCSI Type (00, 01 or 0S)**	**Device Name (rmt0)**	**SCSI Address (0 thru 6)**	**Hdisk Size**
				GBytes

Display information regarding the system's memory by executing:

```
# lscfg -vl mem* | pg
```

The output of the command will be:

```
# lscfg -vl mem* | pg
  DEVICE            LOCATION          DESCRIPTION

  mem0              00-0B             64 MB Memory Card

   Device Specific.(Z3)........07
   EC Level....................21
   Device Specific.(Z0)........00
   Device Specific.(Z1)........00
   Device Specific.(Z2)........00
        Size.......................64

  mem1              00-0C             64 MB Memory Card

   Device Specific.(Z3)........07
   EC Level....................33
   Device Specific.(Z0)........00
   Device Specific.(Z1)........00
   Device Specific.(Z2)........01
   Size.......................64
:
```

Record the information about the size and slot location of IBM-provided memory cards installed in the system. Press the Enter key as needed to scroll forward through the memory information, and return to the command prompt.

Table 13. Memory Configuration	
Memory Slot Location	**Memory Card Size (MB)**
A	
B	
C	
D	
E	
F	
G	
H	

Use the information in the previous table to verify which cards are available to be upgraded. Do not proceed with the system upgrade if the memory requirements are not met.

Get information regarding the adapters by executing the following command:

```
# lsdev -C -c adapter|pg
```

The output of the command will be:

```
# lsdev -C -c adapter|pg
sio0     Available 00-00    Standard I/O Planar
fda0     Available 00-00-0D Standard I/O Diskette Adapter
siokta0  Available 00-00-0K Keyboard/Tablet Adapter
sa0      Available 00-00-S1 Standard I/O Serial Port 1
sa1      Available 00-00-S2 Standard I/O Serial Port 2
scsi0    Available 00-06    SCSI I/O Controller
scsi1    Available 00-07    SCSI I/O Controller
scsi2    Available 00-08    SCSI I/O Controller
slc0     Available 00-00    Serial Optical Link Chip
slc1     Available 00-00    Serial Optical Link Chip
sioma0   Available 00-00-0M Mouse Adapter
ppa0     Available 00-00-0P Standard I/O Parallel Port Adapter
tok0     Available 00-02    Token-Ring High-Performance Adapter (8fc8)
tok1     Available 00-03    Token-Ring High-Performance Adapter (8fc8)
apm0     Available 00-05    4-Port Selectable Portmaster Adapter/A (MPQP)
:
```

Record the slot location, physical name and logical name of the displayed adapters or pipe the output to a printer. Press the Enter key as needed to scroll forward through the adapter information.

Table 14. Adapter Information

Slot Number	Physical Name	Logical Name
1		
2		
3		
4		
5		
6		
7		
8		

The following procedure will help you gather the information that is necessary for configuring the ttys and printers on the multiprocessor as they were on the uniprocessor.

Print the information regarding the asynchronous adapters numbering by executing the following command:

```
# lsdev -C | grep sa | lp
```

The printout of the `lsdev -C | grep sa | lp` command will be:

```
sa0        Available 00-00-S1     Standard I/O Serial Port 1
sa1        Available 00-00-S2     Standard I/O Serial Port 2
sa2        Available 00-04-21     16-Port RAN EIA-232 for 128-Port Adapter
```

Print the information regarding the tty numbering by executing the following command:

`# lsdev -C | grep tty | lp`

The printout of the `lsdev -C | grep tty | lp` command will be:

```
tty0       Available 00-00-S1-00  Asynchronous Terminal
tty1       Available 00-04-21-00  Asynchronous Terminal
tty2       Available 00-04-21-01  Asynchronous Terminal
tty3       Available 00-04-21-02  Asynchronous Terminal
tty4       Available 00-04-21-03  Asynchronous Terminal
tty5       Available 00-04-21-04  Asynchronous Terminal
tty6       Available 00-04-21-05  Asynchronous Terminal
tty7       Available 00-04-21-06  Asynchronous Terminal
```

Print out the information regarding the printer numbering by executing the following command:

`# lsdev -C | grep lp | lp`

The printout of the `lsdev -C | grep lp | lp` command will be:

```
lp0        Available 00-00-S2-00  IBM 3812 Model 2 Page Printer
lp1        Available 00-00-0P-00  Other parallel printer
```

Now that you have the printout of the asynchronous adapters, printers and ttys, you can collate the information, and insert it into the following table.

Table 15 (Page 1 of 2). TTY and Printer Configuration			
Asynchronous Adapters		**TTY and Printer**	
sa Number	**Location**	**TTY / Printer Name**	**Location**
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-

Table 15 (Page 2 of 2). TTY and Printer Configuration			
Asynchronous Adapters		**TTY and Printer**	
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-
	00-		00-

Get information regarding the LPPs that are installed on the system by executing the following command:

```
# lslpp -l|pg
```

The output of the command will be:

```
    Name                    State           Description
  -------------------     -----------     --------------------------------------
 Path: /usr/lib/objrepos
   bos.obj 03.02.00.00     COMMITTED       The Base Operating System
   bosext1.csh.obj 03.02.00.00
                           COMMITTED       C Shell
   bosext1.extcmds.obj 03.02.00.00
                           COMMITTED       Extended Commands
   bosext2.dosutil.obj 03.02.00.00
                           COMMITTED       DOS Utilities
   bosnet.ncs.obj 03.02.00.00
                           COMMITTED       Network Computing System
   bosnet.snmpd.obj 03.02.00.00
                           COMMITTED       Simple Network Management Protocol
                                           Daemon (Agent)
   bosnet.tcpip.obj 03.02.00.00
                           COMMITTED       TCPIP Applications
   xlccmp.obj 01.03.00.00
                           APPLIED         AIX XL C Compiler
 :
```

Record the information regarding the licensed program products (LPPs) that are installed on the system or pipe the ouput of lslpp -l to a printer.

Table 16 (Page 1 of 2). Licensed Program Products		
Version Number	**LPP Name**	**Product Name**
		AIX
		AIX
		AIX
		AIX
		AIX
		AIX

Table 16 (Page 2 of 2). Licensed Program Products		
Version Number	**LPP Name**	**Product Name**
		AIX
		AIX
		AIX
		AIX
		AIX
		AIX

4.4.4 Backing Up the UP System

At this step, you must back up your entire uniprocessor system and create a bootable tape. User-defined volume groups, raw disks, and raw Logical Volumes must be backed up as well.

For more information on how to back up your entire AIX V3.2.5 system, we recommend that you refer to 1.3, "Backing Up Your AIX V3.2.5 System" on page 30 in this redbook.

4.4.5 Migrating AIX from V3.2.5 to V4.1

We recommend that before migrating AIX to AIX V4.1, you carefully read Chapter 1, "Migrating AIX from V3.2.5 to V4.1" on page 3 and more specifically 1.4, "Migrating AIX" on page 33 for instructions and recommendations on how to migrate AIX.

After the system has been upgraded to AIX Version 4.1 and tested at this level, you should make another system backup. Refer to 1.6, "Backing Up your AIX V4.1 System" on page 57 for more information on how to back up your AIX V4.1 system.

After testing and backing up have been successfully completed, you are ready to carry on your upgrade to the SMP.

Note: If the system has been used between the backup of the system and the continuation of the upgrade, you will have to make another backup of rootvg (or at least user data), user-defined volume groups and raw logical volumes, if any.

4.4.6 Installing the Required Device Drivers and Filesets

The following packages and/or filesets are required by an SMP system as a minimum for a successful boot.

- bos.rte.mp - MP kernel

- devices.mca.fed9 - Standard I/O device driver

- devices.mca.8efc - 16-bit SCSI I/O controller device driver

- devices.rs6ksmp.base - MP base system device support

You must also install support for the additional adapters and I/O devices that have been ordered with the upgrade, if any. You will also be able to install additional device support after the upgrade, if necessary, with the following command:

```
# cfgmgr -i /dev/rmt0
```

where /dev/rmt0 is your installation device.

Note: Check the hardware upgrade documentation to find out if any additional adapters or I/O devices have been ordered; then check the additional hardware against the device drivers list provided in Appendix G, "AIX V4.1 Device Driver Packages" on page 319.

We also recommend that you install, if they are not already installed after the migration to AIX V4.1, the following filesets:

- bos.diag.rte - This fileset contains commands, like mpcfg and keycfg, related to the SystemGuard service processor.

- bos.sysmgt.serv_aid - This fileset contains error logging and service aids for SystemGuard daemons survd and mirrord.

- bos.sysmgt.sysbr - This fileset contains back up utilities such as mksysb and savevg.

In order to check to see if a fileset is already installed or not, enter the following command:

```
# lslpp -L | grep <fileset_name>
```

where <fileset_name> is the name of the fileset you are looking for.

Note!

The upgrade kit includes a Model Upgrade Utilities diskette containing a shell script called mp_prep that will automatically install the required filesets, create the MP kernel links and the MP boot image. You will find the content of the mp_prep script at the end of this chapter.

The procedures below can be followed instead. The intent here is to help you in understanding the way a UP to SMP upgrade works.

To install the multiprocessor fileset and the minimum device drivers that will enable you to boot your SMP system, you can use the following procedure.

4.4.6.1 Multiprocessor Fileset Installation
Use the following procedure to install the multiprocessor filesets.

1. Load the AIX media into the appropriate media device.

2. Log on as root.

   Note: If you are using an ASCII terminal as your output device set the terminal environment by entering the TERM=xxxxxxx command (for example; TERM=ibm3151) and export the terminal environment.

   ```
   #export TERM=ibm3151
   #
   ```

3. Enter smitty install_latest

```
                    Install Software Products at Latest Level

     Type or select a value for the entry field.
     Press Enter AFTER making all desired changes.

                                                         [Entry Fields]
     * INPUT device / directory for software          []        +



     F1=Help          F2=Refresh        F3=Cancel           F4=List
     F5=Reset         F6=Command        F7=Edit             F8=Image
     F9=Shell         F10=Exit          Enter=Do
```
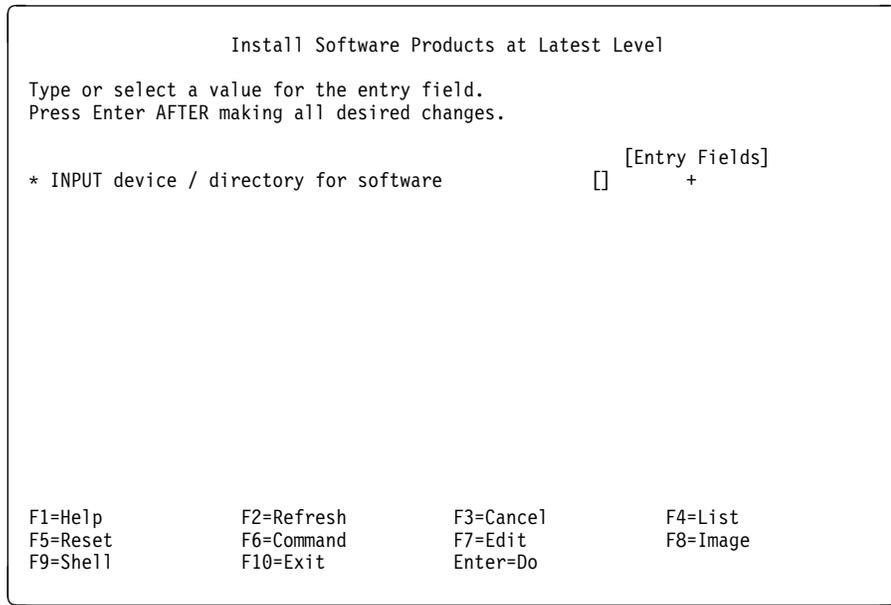
4. At the Install Software Products at Latest Level menu, the selection INPUT
   device / directory for software will be highlighted. Press the F4 key to list
   the input device options.

5. Select the media device that has the AIX media installed in it by using the
   arrow keys to highlight your choice; then press the Enter (or Return) key.

6. The screen should now display the Install Software Products at Latest Level
   menu again, this time with the selection SOFTWARE to install highlighted.
   Press the F4 key again to list the software packages to choose from.

```
                    Install Software Products at Latest Level

     Type or select values in entry fields.
     Press Enter AFTER making all desired changes.

                                                         [Entry Fields]
     * INPUT device / directory for software          /dev/cd0
     * SOFTWARE to install                            []        +
       PREVIEW only? (install operation will NOT occur)   no              +
       COMMIT software updates?                       yes             +
       SAVE replaced files?                           no              +
       ALTERNATE save directory                       []
       AUTOMATICALLY install requisite software?      yes             +
       EXTEND file systems if space needed?           yes             +
       OVERWRITE same or newer versions?              no              +
       VERIFY install and check file sizes?           no              +
       Include corresponding LANGUAGE filesets?       yes             +
       DETAILED output?                               no              +



     F1=Help          F2=Refresh        F3=Cancel           F4=List
     F5=Reset         F6=Command        F7=Edit             F8=Image
     F9=Shell         F10=Exit          Enter=Do
```

7. When the SOFTWARE to install menu appears, type / (the forward slash key)
   to start the Find menu.

8. At the Find menu, type the following software package name to be used as the
   search pattern:

   bos.rte.mp

and press the Enter (or Return) key.

9. `bos.rte.mp` should now be highlighted. Press the F7 key to select this package for downloading from the AIX media to the system's disk drive.

10. Press the Enter (or Return) key now that your selections have been made.

11. Press the Enter (or Return) key to start the download process.

12. When the Are You Sure? submenu pops up, press the Enter (or Return) key again.

13. The Command Status menu should be displayed in the upper left-hand corner of the screen. The command indicator should display `running` to show that the software packages are being downloaded.

14. When the command indicator changes from `running` to `OK`, the download has completed. Press the F10 key to exit when the download is completed.

15. Leave the installation media inserted in the media drive.

4.4.6.2 Additional Device Driver Installation
Install certain device software packages needed for booting the new model by doing the following:

1. With the AIX media still installed, enter:

   `# smitty devinst`

2. The menu option `INPUT device / directory for software` should now be highlighted. Press the F4 key to list the input device choices.

```
                      Install Additional Device Software

 Type or select a value for the entry field.
 Press Enter AFTER making all desired changes.

                                                   [Entry Fields]
 * INPUT device / directory for software          []         +













 F1=Help              F2=Refresh         F3=Cancel          F4=List
 F5=Reset             F6=Command         F7=Edit            F8=Image
 F9=Shell             F10=Exit           Enter=Do
```
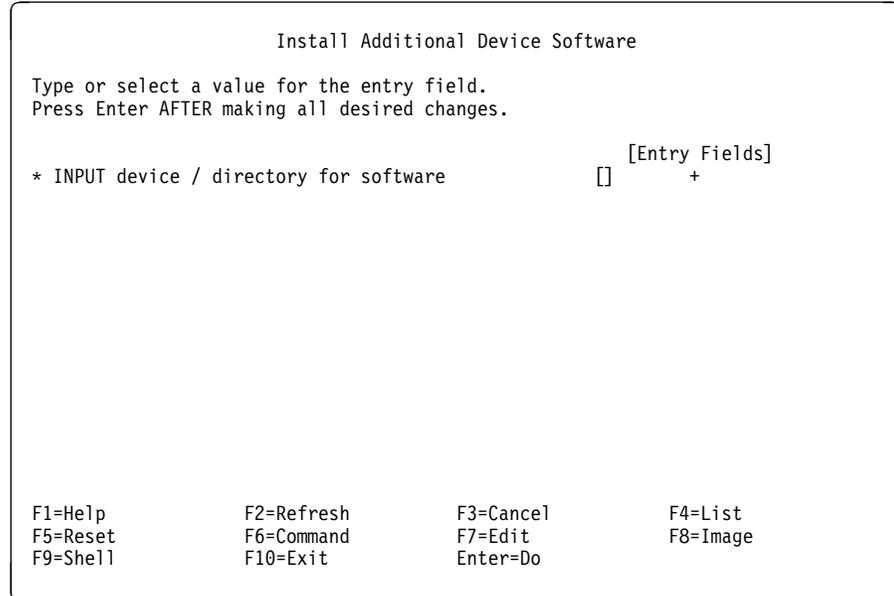
3. Select the media device that has the AIX media installed in it by using the arrow keys to highlight your choice; then press the Enter (or Return) key.

4. The screen should now display the Install Additional Device Software menu again, this time with the selection `SOFTWARE to install` highlighted. Press the F4 key again to list the software packages to choose from.

5. When the SOFTWARE to install menu appears, type / (the forward slash key) to start the Find menu.

6. At the Find menu type the following software package name to be used as the search pattern,

```
devices.mca.fed9
```

and press the Enter (or Return) key. The above software package should now be highlighted. Press the F7 key to select this package for downloading from the AIX media to the system's disk drive.

7. Next type / (the forward slash key) again to search for another software package. At the Find menu, type the following software package name to be used as the search pattern,

```
devices.mca.8efc
```

and press the Enter (or Return) key. The above software package should now be highlighted. Press the F7 key to select this package for downloading from the AIX media to the system's disk drive.

8. Next type / (the forward slash key) again to search for another software package. At the Find menu, type the following software package name to be used as the search pattern,

```
devices.rs6ksmp.base
```

and press the Enter (or Return) key. The above software package should now be highlighted. Press the F7 key to select this package for downloading from the AIX media to the system's disk drive. Press the Enter (or Return) key now that your selections have been made. Press the Enter (or Return) key to start the download process.

9. When the Are You Sure? submenu pops up, press the Enter (or Return) key again. The Command Status menu should be displayed. In the upper left-hand corner of the screen, the command indicator should display `running` to show that the software packages are being downloaded.

10. When the command indicator changes from `running` to `OK`, the download is completed. Press the F10 key to exit when the download is completed.

11. Remove the AIX media when the system has finished installing the missing software package(s).

This is the minimum software that needs to be loaded in order to support the new devices on the SMP system. If there are additional features that have been ordered for the SMP system and have been installed into the new chassis by the shipping plant, then the additional device support should now be loaded by the above procedure (steps 1 to 10) before you continue with the upgrade procedure.

After the device support has been loaded, you must now link the multiprocessor kernel to the /unix file. The following procedure will show you how to accomplish this.

4.4.7 Creating the MP Kernel Links and Boot Image

There are two links that must be made to the multiprocessor (MP) kernel. Establish links to the AIX multiprocessor kernel by doing the following:

1. Ensure you are in the root directory by entering:

```
# cd /
```

2. Remove the current link to the UP kernel by entering:

```
# unlink unix
```

3. Add the new link to the MP kernel by entering:

```
# ln -s /usr/lib/boot/unix_mp unix
```

4. Verify the new link by entering:

```
# ls -l unix
```

5. The last portion of the displayed response should be

```
# unix -> /usr/lib/boot/unix_mp
```

if the link is established correctly. If the link is not correctly established, then repeat the steps 1 through 5.

6. Change directories to establish another MP link by entering:

```
# cd /usr/lib/boot
```

7. Remove the current link to the UP kernel in this directory by entering:

```
# unlink unix
```

8. Add the new link to the MP kernel by again entering:

```
# ln -s /usr/lib/boot/unix_mp unix
```

9. Verify the new link in this directory by entering:

```
# ls -l unix
```

10. The last portion of the displayed response should be

```
# unix -> /usr/lib/boot/unix_mp
```

if the link is established correctly. If the link is not correctly established, then repeat the steps 6 through 10.

11. Make the multiprocessor kernel the default kernel by executing the following command:

```
# bosboot -a -d/dev/hdiskx -k /usr/lib/boot/unix_mp -T rs6ksmp
```

where `hdiskx` is the disk containing the boot logical volume that you recorded previously.

Note: The `-k` option is optional since /unix is already linked to /usr/lib/boot/unix_mp. The option -T allows you to specify the platform type. Since you build a boot image for the SMP system, you must specify the SMP platform type, rs6ksmp. If this option is not specified, you will create an MP boot image for the UP system because the `bosboot` command takes, by default, the platform type of the machine on which you are creating the boot image.

> ── **Attention!** ──────────────────────────────
>
> After the boot image is created using the `-T rs6ksmp` option, the uniprocessor will not be able to boot unless you recreate a UP boot image or an MP image without the `-T` option.
>
> If you try to boot your UP system, you will get a 554 code on the LED display. You will need to boot on your backup tape or on a product CD-ROM or tape, enter into a maintenance shell and recreate the appropriate boot image.

4.4.8 Modifying the ODM Configuration

After the additional filesets and device drivers have been installed, and the `bosboot` command has completed, you must execute one of the shell scripts (`mach.upg`) included in the Model Upgrade Utilities diskette.

Note: The content of the `mach.upg` shell script is provided at the end of this chapter for your convenience

This program will modify the uniprocessor's ODM by deleting entries in the ODM such as all standard I/O adapters, the standard I/O planar device (sio0), all devices related to the serial optical link chips, all of the system devices (ioplanar0, sysplanar0), memory, and finally the system device itself sys0. This allows you to wipe out the ODM before creating the system backup that you are going to restore on the SMP system.

If it is not alredy done, restore in /tmp the content of the Model Upgrade Utilities diskette by using the following procedure:

- Change to the /tmp directory:

  `# cd /tmp`

- Insert the diskette into the diskette drive, and enter the following command:

  `# restore -xvq`

  Note: /dev/rfd0 is the default device for the restore command. The file that is restored to the /tmp directory is called mach.upg.

- Execute the `mach.upg` shell script with the following command:

  `# ./mach.upg`

- Answer `YES` to the questions that appear on the screen.

  Note: After the `mach.upg` program has updated the system files, the command prompt is displayed. Do not proceed with the next step until the command prompt is displayed.

- Change directory to /etc, and remove any reconfig files:

  `# rm /tmp/reconfig*`

  Note: Do not be concerned if the files /tmp/reconfig1 and /tmp/reconfig2 do not exist in /tmp.

4.4.9 Creating an MP Backup Tape

After the `mach.upg` script has completed and you have removed the reconfig files in /tmp, you are now ready to create a bootable backup tape of the modified rootvg.

Use the SMIT backup utility to achieve this.

`# smitty mksysb`

Note: At this step, it is not necessary to again back up user-defined volume groups.

4.4.10 Migrating the Hardware

This upgrade will swap the down-level chassis with the new chassis. All supported adapters, memory and physical volumes will simply be moved to the new chassis.

Each upgrade is specific. A detailed procedure on how to migrate the hardware will be provided with the upgrade kit. We recommend that you follow it carefully.

4.4.11 Restoring the System Backup on the SMP

After the hardware has been upgraded, it is necessary to recover rootvg and the user-defined volume groups and/or raw logical volumes. The following section gives a step-by-step procedure on how to accomplish this.

1. Label the cables that plug into the asynchronous adapters, and unplug these cables from the asynchronous adapters.

   Note: The reason these cables must be disconnected is that you do not want the asynchronous 16-port RANs to be configured after the system backup has been restored. This will enable you to restore the printers and ttys to their original adapters later in this procedure.

2. Plug the external options and devices into the power outlets.

3. Ensure the Main Power switch, which is located at rear of the system, is off ("0" position). Plug the system unit power cord into the power outlet.

4. Turn on all external options and devices.

5. Set the key mode switch to the Service position (the wrench symbol).

6. Turn the system Main Power switch on. The words `Stand-By` should appear on the operator panel display on the front of the system.

Note: The following procedure is used to modify the stored NVRAM checksum value. This forces the SystemGuard service processor to poll the actual configuration. The system should start out in the Stand-By mode.

1. If the console prompt sign is NOT present, press Enter (or return).
2. At the ASCII terminal prompt (>), type the following:

   ```
   > sbb
   ```

   and press the Enter (or Return) key. This will bring up the STAND-BY MENU.

```
   STAND-BY MENU :  rev 16.00

0 Display Configuration
1 Set Flags
2 Set Unit Number
3 Set Configuration
4 SSbus Maintenance
5 I2C Maintenance








Select(x:exit): 0
```

3. Type 4, and press the Enter (or Return) key to open the SSbus Maintenance menu.
4. Now type 01, and press the Enter (or Return) key to select the **Write nvRam** option.

```
    SSbus Maintenance

   Command     Agent             Start-end Address

00 Read        nvram             0x4000-0x7ffff
01 Write       nvram
02 Read        line S1           0x000207f0
03 Read        line S2           0x000507f0
04 Read        TOD               0x38000-0x3803e
05 Read        TOD
06 Read        TOD




Select (x:exit):
```

5. When the system displays `Start Address` at the bottom of the screen, type:

   `40233`

   and press the Enter (or Return) key.
6. At the next screen, which will display `00040233 00`, type:

   `FF`

   and press the Enter (or Return) key.
7. At the next screen, which will display `40234 00`, type:

   `.` (a period)

   and press the Enter (or Return) key.
8. Type x, and press the Enter (or Return) key to exit.
9. Once again, type x, and press the Enter (or Return) key to exit.

10. The ASCII terminal should have a blank screen.
11. Leave the system key in the Service position.
12. Push in the power button located on the operator panel.
13. Insert the mksysb tape that you made earlier into the tape drive.
14. Wait for the power on tests to complete.
15. The console will now display the Maintenance Menu.

```
                        Maintenance Menu

   0> Display Configuration
   1> Display Bump Error Log
   2> Enable Service Console
   3> Disable Service Console
   4> Reset
   5> Power Off
   6> System Boot
   7> Off-Line Tests
   8> Set Parameters
   9> Set National Language




   Select :
```

16. Select option **6> System Boot** from the Maintenance Menu.
17. The System Boot menu will now be displayed.

```
                        System Boot

   0 > Boot From List

   1 > Boot From Network

   2 > Boot From SCSI Device









   Select [x:exit] :
```

18. Select option **0> Boot from List**.

   The following will now be displayed on the console:

```
    processor 0 on ipl start
    {253}
<<299>>
```

After approximately 10 minutes, you will be asked to define the system console. You will then be asked to select the language that you require.

Next to be displayed on the console will be the Installation and Maintenance Menu.

```
                  Welcome to Base Operating System
                     Installation and Maintenance

 Type the number of your choice and press Enter. Choice is indicated by >>

    >>> 1 Start Install Now with Default Settings
         2 Change/Show Installation Settings and Install
         3 Start Maintenance Mode for System Recovery




                               +--------------------------------------------
                               | An invalid disk (00-08-00-0,0 was
                               | specified in the location field of the
                               | data file
     88 Help ?                 |
     99 Previous Menu

    >>> Choice [1] :
```

An error will occur stating that there is an invalid disk. Ignore this message.
19. Select option **3> Start Maintenance Mode for System Recovery**.
20. At the Maintenance Menu, select option **4> Install from a System Backup**.

```
                     Maintenance

   Type the number of your choice and press Enter.


   >>> 1 Access a Root Volume Group
        2 Copy a System Dump to Removable Media
        3 Access Advanced Maintenance Functions
        4 Install from a System Backup









     88 Help ?
     99 Previous Menu

    >>> Choice [1] :
```

Next, the Choose Tape Drive menu will be displayed.

```
                  Choose Tape Drive

   Type the number of the tape drive containing the system backup to be
   installed and press Enter.

        Tape Drive                        Path Name
   >>> 1 tape/scsi/8mm                     /dev/rmt0
       2 tape/scsi/525mb                   /dev/rmt1






   88 Help ?
   99 Previous Menu

   >>> Choice [1] :
```

21. Select the tape drive that contains the mksysb.

    After you have selected the tape drive, the console will revert back to the
    Installation and Maintenance menu.

22. Select option **1 Start Install Now with Default Settings** on the Installation and
    Maintenance menu.

    Next, an Installation Warning menu will be displayed:

```
                  Installation Warning

   WARNING: Base Operating System Installation may destroy or impair
            recovery of data. Before installing, you should backup
            your system.

   >>> 1 Continue with Install








   88 Help ?
   99 Previous Menu

   >>> Choice [1] :
```

23. Select option **1 Continue with install**, and ignore the warning message as
    there is nothing loaded onto the physical volumes. When the Installing Base
    Operating System screen is displayed on the console, your system is now busy
    restoring the mksysb.

```
      Installing Base Operating System

  Turn the system key to the NORMAL position any time before the
  installation ends.




    Please Wait ....




    Approximate                Elapsed time
  % tasks complete             (in minutes)

      10                          2       (job in progress)
```
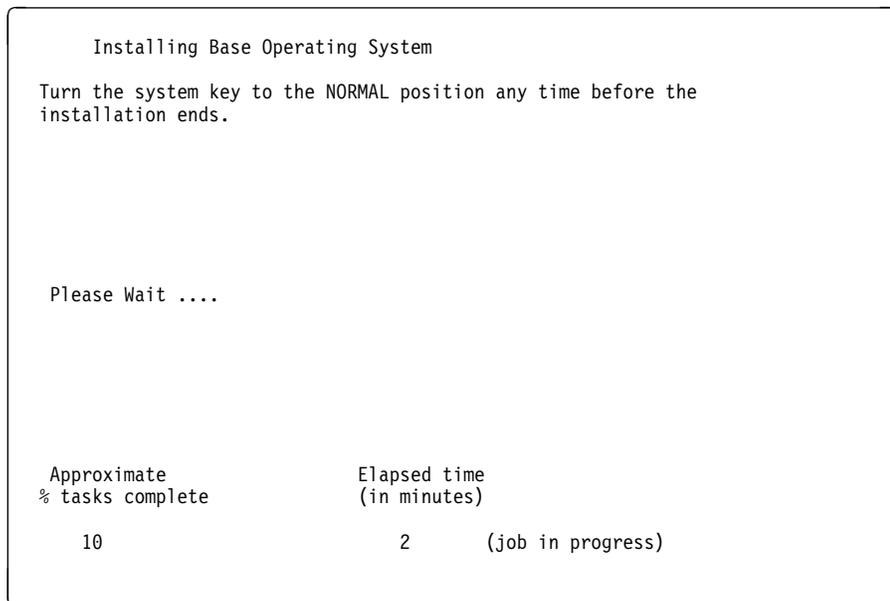
At this time, turn the system key from the SERVICE position to the NORMAL position.

24. When you have a logon prompt, log onto the system as root.
25. Run diagnostics by entering:

    # diag

26. The Diagnostic Operating Instructions menu should be displayed.

    Note:

    If the menu does not appear, run diagnostics from tape or CD-ROM. This will require rebooting the system in the Service mode.

    You may be required to initialize the terminal when running diagnostics.

27. Press the Enter key.
28. Select **Advanced Diagnostic Routines** on the Function Selection menu.
29. Select **System Verification** on the Diagnostic Mode Selection menu.

    Run diagnostics on the base system and all devices on the Advanced Diagnostic Selection menu.

30. After you have tested the system and all of the tests have passed, exit out of diagnostics (F10=EXIT).
31. After all the tests have completed, you are now ready to reload the user-defined volume groups and raw logical volumes.

    Note: User-defined volume groups and raw logical volumes will have to be reloaded if the physical volumes that they were created on were not migrated from the uniprocessor system to the multiprocessor system.

    Recreate and reload the user-defined volume groups and raw logical volumes from the backups that you made earlier.

4.4.12 Reconfiguring TTYs and Printers

After the MP system backup has been restored, you must reconfigure the ttys and printers.

- Reconnect the cables for the asynchronous adapters that you disconnected earlier, and switch on all the connected asynchronous nodes and devices.

  Note: Make sure that the cables are inserted into the correct location on the asynchronous adapters.

- At the command prompt, execute the following command:

  ```
  # cfgmgr -v
  ```

  Attention!

  We assume in the following procedure that the uniprocessor had an ASCII terminal connected and defined on S1 (tty0 on sa0).

  If the uniprocessor had a graphical display and no ASCII terminal configured on S1, the procedure is slightly different.

  This will configure all of the asynchronous nodes and assign sa numbers to them.

- In /tmp, you will find two (2) files that can be used to reconfigure the printers and ttys. The files are:

  - /tmp/reconfig1
  - /tmp/reconfig2

- Copy /tmp/reconfig1 to /tmp/reconfig1.original:

  ```
  # cp /tmp/reconfig1 /tmp/reconfig1.original
  ```

  The reconfig1 file will look similar to the following:

  ```
  #!/bin/ksh
  export TALLY=
  cmd_1='/etc/methods/define -l tty1 -c tty -s rs232 -t tty -p sa2 -w 0 '
  cmd_2='/etc/methods/cfgtty -l tty1 '
  cmd_3='/etc/methods/define -l tty2 -c tty -s rs232 -t tty -p sa2 -w 1 '
  cmd_4='/etc/methods/cfgtty -l tty2 '
  cmd_5='/etc/methods/define -l tty3 -c tty -s rs232 -t tty -p sa2 -w 2 '
  cmd_6='/etc/methods/cfgtty -l tty3 '
  cmd_7='/etc/methods/define -l tty4 -c tty -s rs232 -t tty -p sa2 -w 3 '
  cmd_8='/etc/methods/cfgtty -l tty4 '
  cmd_9='/etc/methods/define -l tty5 -c tty -s rs232 -t tty -p sa2 -w 4 '
  cmd_10='/etc/methods/cfgtty -l tty5 '
  cmd_11='/etc/methods/define -l tty6 -c tty -s rs232 -t tty -p sa2 -w 5 '
  cmd_12='/etc/methods/cfgtty -l tty6 '
  cmd_13='/etc/methods/define -l tty7 -c tty -s rs232 -t tty -p sa2 -w 6 '
  cmd_14='/etc/methods/cfgtty -l tty7 '
  cmd_15='/etc/methods/define -l tty8 -c tty -s rs232 -t tty -p sa2 -w 7 '
  cmd_16='/etc/methods/cfgtty -l tty8 '
  cmd_17='/etc/methods/define -l tty9 -c tty -s rs232 -t tty -p sa2 -w 8 '
  cmd_18='/etc/methods/cfgtty -l tty9 '
  cmd_19='/etc/methods/define -l tty10 -c tty -s rs232 -t tty -p sa2 -w 9 '
  cmd_20='/etc/methods/cfgtty -l tty10 '
  cmd_21='/etc/methods/define -l tty11 -c tty -s rs232 -t tty -p sa2 -w 10 '
  cmd_22='/etc/methods/cfgtty -l tty11 '
  cmd_23='/etc/methods/define -l tty12 -c tty -s rs232 -t tty -p sa2 -w 11 '
  ```

```
cmd_24='/etc/methods/cfgtty -l tty12 '
cmd_25='/etc/methods/define -l tty13 -c tty -s rs232 -t tty -p sa2 -w 12 '
cmd_26='/etc/methods/cfgtty -l tty13 '
cmd_27='/etc/methods/define -l tty14 -c tty -s rs232 -t tty -p sa2 -w 13 '
cmd_27='/etc/methods/cfgtty -l tty14 '
cmd_29='/etc/methods/define -l tty15 -c tty -s rs232 -t tty -p sa2 -w 14 '
cmd_30='/etc/methods/cfgtty -l tty15 '
cmd_31='/etc/methods/define -l tty16 -c tty -s rs232 -t tty -p sa2 -w 15 '
cmd_32='/etc/methods/cfgtty -l tty16 '
cmd_33='chdev -l tty0
        -a login=enable
        -a term=ibm3151'
cmd_34='chdev -l tty1
        -a login=enable
        -a term=ibm3151'
cmd_35='chdev -l tty2
        -a login=enable
        -a term=ibm3151'
cmd_36='chdev -l tty3
        -a login=enable
        -a term=ibm3151'
cmd_37='chdev -l tty4
        -a login=enable
        -a term=ibm3151'
cmd_38='chdev -l tty5
        -a login=enable
        -a term=ibm3151'
cmd_39='chdev -l tty6
        -a login=enable
        -a term=ibm3151'
cmd_40='chdev -l tty7
        -a login=enable
        -a term=ibm3151'
cmd_41='chdev -l tty8
        -a login=enable
        -a term=ibm3151'
cmd_42='chdev -l tty9
        -a login=enable
        -a term=ibm3151'
cmd_43='chdev -l tty10
        -a login=enable
        -a term=ibm3151'
cmd_44='chdev -l tty11
        -a login=enable
        -a term=ibm3151'
cmd_45='chdev -l tty12
        -a login=enable
        -a term=ibm3151'
cmd_46='chdev -l tty13
        -a login=enable
        -a term=ibm3151'
cmd_47='chdev -l tty14
        -a login=enable
        -a term=ibm3151'
cmd_48='chdev -l tty15
        -a login=enable
        -a term=ibm3151'
cmd_49='chdev -l tty16
```

```
                    -a login=enable
                    -a term=ibm3151'
        num_cmds=50
        i=0
        while [ $i -lt $num_cmds ]
        do
            eval \$cmd_$i > /dev/null
            if [ $? -ne 0 ]
            then
                TALLY="$TALLY$(eval echo \$cmd_$i)\n"
            fi
            (( i+=1 ))
        done
        if [ -n "$TALLY" ]
        then
            echo Machine not identical to previous configuration.
            echo The following items failed:
            echo $TALLY
        fi
```

- Print the information regarding the asynchronous adapters numbering by executing the following command:

`# lsdev -C | grep sa | lp`

The printout of the `lsdev -C | grep sa | lp` command will be:

```
sa0        Available 00-00-S1      Standard I/O Serial Port 1
sa1        Available 00-00-S2      Standard I/O Serial Port 2
sa2        Available 00-00-S3      Standard I/O Serial Port 2
sa3        Available 00-04-21      16-Port RAN EIA-232 for 128-Port Adapter
```

- Using the table of the ttys, printers and sa numbers that you created earlier, map the sa numbers from the uniprocessor system to the sa numbers of the multiprocessor. Following is a diagram of a typical uniprocessor asynchronous port configuration that has had its configuration mapped to a multiprocessor after the `cfgmgr` command was executed.
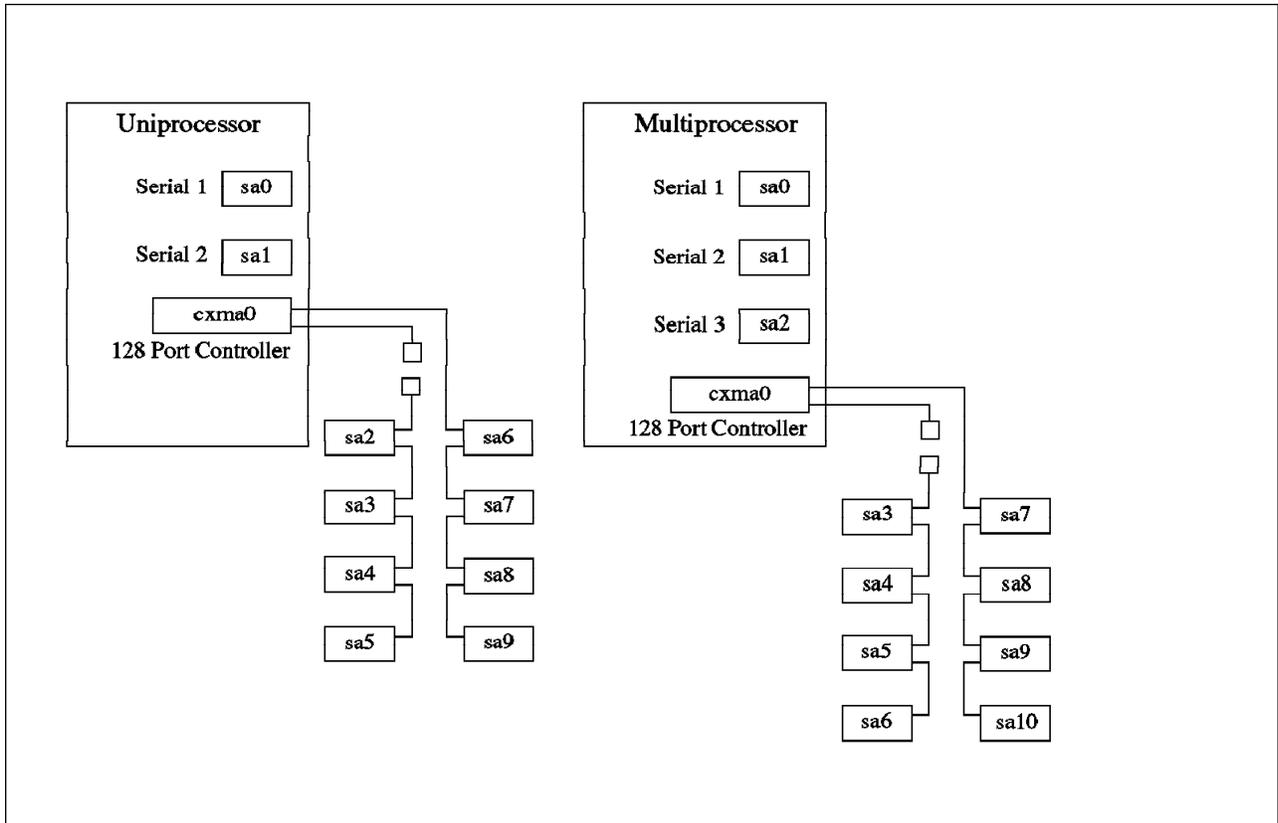
Figure 47. Asynchronous Port Configurations

- Following is a sample table that maps the sa numbers, ttys and printers from the uniprocessor configuration to the multiprocessor configuration.

Table 17. Asynchronous Port Mapping							
Uniprocessor				Multiprocessor			
sa0	00-00-s1	tty0	00-00-s1-00	sa0	00-00-s1	tty0	00-00-s1-00
sa1	00-00-s2	lp0	00-00-s2-00	sa1	00-00-s2	lp0	00-00-s2-00
sa2	00-01-21	tty1	00-01-21-00	sa2	00-00-s3		
		tty2	00-01-21-01	sa3	00-04-21	tty1	00-04-21-00
		tty3	00-01-21-02			tty2	00-04-21-01
		tty4	00-01-21-03			tty3	00-04-21-02
		tty5	00-01-21-04			tty4	00-04-21-03
		tty6	00-01-21-05			tty5	00-04-21-04
		tty7	00-01-21-06			tty6	00-04-21-05
		tty8	00-01-21-07			tty7	00-04-21-06
		tty9	00-01-21-08			tty8	00-04-21-07
		tty10	00-01-21-09			tty9	00-04-21-08
		tty11	00-01-21-10			tty10	00-04-21-9
		tty12	00-01-21-11			tty11	00-04-21-10
		tty13	00-01-21-12			tty12	00-04-21-11
		tty14	00-01-21-13			tty13	00-04-21-12
		tty15	00-01-21-14			tty14	00-04-21-13
		tty16	00-01-21-15			tty15	00-04-21-14
						tty16	00-04-21-15

- Now that you have a map of the sa numbers, ttys and printers. You must substitute the parameter `-p sa2` to `-p sa3`. Edit the /tmp/reconfig1 file to look similar to the following:

```
/etc/methods/define -l tty1 -c tty -s rs232 -t tty -p sa3 -w 0
/etc/methods/cfgtty -l tty1
/etc/methods/define -l tty2 -c tty -s rs232 -t tty -p sa3 -w 1
/etc/methods/cfgtty -l tty2
/etc/methods/define -l tty3 -c tty -s rs232 -t tty -p sa3 -w 2
/etc/methods/cfgtty -l tty3
/etc/methods/define -l tty4 -c tty -s rs232 -t tty -p sa3 -w 3
/etc/methods/cfgtty -l tty4
/etc/methods/define -l tty5 -c tty -s rs232 -t tty -p sa3 -w 4
/etc/methods/cfgtty -l tty5
/etc/methods/define -l tty6 -c tty -s rs232 -t tty -p sa3 -w 5
/etc/methods/cfgtty -l tty6
/etc/methods/define -l tty7 -c tty -s rs232 -t tty -p sa3 -w 6
/etc/methods/cfgtty -l tty7
/etc/methods/define -l tty8 -c tty -s rs232 -t tty -p sa3 -w 7
/etc/methods/cfgtty -l tty8
/etc/methods/define -l tty9 -c tty -s rs232 -t tty -p sa3 -w 8
/etc/methods/cfgtty -l tty9
/etc/methods/define -l tty10 -c tty -s rs232 -t tty -p sa3 -w 9
/etc/methods/cfgtty -l tty10
/etc/methods/define -l tty11 -c tty -s rs232 -t tty -p sa3 -w 10
/etc/methods/cfgtty -l tty11
/etc/methods/define -l tty12 -c tty -s rs232 -t tty -p sa3 -w 11
/etc/methods/cfgtty -l tty12
/etc/methods/define -l tty13 -c tty -s rs232 -t tty -p sa3 -w 12
/etc/methods/cfgtty -l tty13
/etc/methods/define -l tty14 -c tty -s rs232 -t tty -p sa3 -w 13
/etc/methods/cfgtty -l tty14
/etc/methods/define -l tty15 -c tty -s rs232 -t tty -p sa3 -w 14
/etc/methods/cfgtty -l tty15
/etc/methods/define -l tty16 -c tty -s rs232 -t tty -p sa3 -w 15
/etc/methods/cfgtty -l tty16
chdev -l tty0 -a login=enable -a term=ibm3151
chdev -l tty1 -a login=enable -a term=ibm3151
chdev -l tty2 -a login=enable -a term=ibm3151
chdev -l tty3 -a login=enable -a term=ibm3151
chdev -l tty4 -a login=enable -a term=ibm3151
chdev -l tty5 -a login=enable -a term=ibm3151
chdev -l tty6 -a login=enable -a term=ibm3151
chdev -l tty7 -a login=enable -a term=ibm3151
chdev -l tty8 -a login=enable -a term=ibm3151
chdev -l tty9 -a login=enable -a term=ibm3151
chdev -l tty10 -a login=enable -a term=ibm3151
chdev -l tty11 -a login=enable -a term=ibm3151
chdev -l tty12 -a login=enable -a term=ibm3151
chdev -l tty13 -a login=enable -a term=ibm3151
chdev -l tty14 -a login=enable -a term=ibm3151
chdev -l tty15 -a login=enable -a term=ibm3151
chdev -l tty16 -a login=enable -a term=ibm3151
```

- When you have finished editing the /tmp/reconfig1 file, execute the file with the following command:

```
# ./tmp/reconfig1
```

This command will reconfigure the ttys and printers onto the controllers and asynchronous nodes on the multiprocessor as they were on the uniprocessor.

Attention!

If, for example, the uniprocessor does not have any terminal connected and configured on its native serial port S1 because a graphical display was used as a console and you migrate to a multiprocessor that requires an ASCII console, your first tty defined on your 16-port RAN might not be configured on the multiprocessor.

Below is an example with a different asynchronous port mapping. The uniprocessor does not have any tty configured on the native serial ports and the multiprocessor requires a tty0 as a console.

Table 18. Asynchronous Port Mapping							
Uniprocessor				**Multiprocessor**			
sa0	00-00-s1		00-00-s1-00	sa0	00-00-s1	tty0	00-00-s1-00
sa1	00-00-s2		00-00-s2-00	sa1	00-00-s2		00-00-s2-00
sa2	00-01-21	tty0	00-01-21-00	sa2	00-00-s3		
		tty1	00-01-21-01	sa3	00-04-21	**tty16**	00-04-21-00
		tty2	00-01-21-02			tty1	00-04-21-01
		tty3	00-01-21-03			tty2	00-04-21-02
		tty4	00-01-21-04			tty3	00-04-21-03
		tty5	00-01-21-05			tty4	00-04-21-04
		tty6	00-01-21-06			tty5	00-04-21-05
		tty7	00-01-21-07			tty6	00-04-21-06
		tty8	00-01-21-08			tty7	00-04-21-07
		tty9	00-01-21-09			tty8	00-04-21-08
		tty10	00-01-21-10			tty9	00-04-21-9
		tty11	00-01-21-11			tty10	00-04-21-10
		tty12	00-01-21-12			tty11	00-04-21-11
		tty13	00-01-21-13			tty12	00-04-21-12
		tty14	00-01-21-14			tty13	00-04-21-13
		tty15	00-01-21-15			tty14	00-04-21-14
						tty15	00-04-21-15

After running the modified /tmp/reconfig1 file, you will notice that tty0 has been configured on the first native serial port, named sa0. No tty will be configured on the first port (00) of the 16-port RAN.

You will then need to retrieve the original tty0 configuration, and recreate with SMIT a tty16 at location 00-04-21-00 (port 00 of the 16-port RAN).

4.4.13 Post-Upgrade Tasks

Once all your terminals and printers are properly configured, some additional tasks might need to be performed. This list is not exhaustive:

- Check to see if all the device drivers you need have been installed. Put your product CD-ROM or tape in the installation media, and run the following command:

  `# cfgmgr -i /dev/<device_name>`

  where `<device_name>` is your installation device.

- Shut down the system.

- Check that the key is in Normal and reboot.

- Check again your overall device configuration.

- Install additional products.

- Install license keys for all the products that require them. Refer to Chapter 2, "Installing iFOR/LS" on page 71 in this redbook to install license keys.

- Test your entire system.

- Back up your system.

4.5 Migrating an AIX V4.1/UP to an AIX V4.1.3/SMP

If your uniprocessor is already running an early version of AIX V4.1 (AIX V4.1.0 or AIX V4.1.1) which does not support SMP systems, you will need first to upgrade AIX to AIX V4.1.3 or AIX V4.1.4. We recommend that you go directly to AIX V4.1.3 or AIX V4.1.4 and not to AIX V4.1.2, if possible. Refer to *AIX V4.1.3 Release Notes* or *AIX V4.1.4 Release Notes* before upgrading your system.

Once your system is running a version of AIX that supports your SMP system, the procedure is the same as previously described.

4.6 Upgrade Utility Shell Scripts

You will find below two shell scripts that are provided with a UP to SMP upgrade in the Model Upgrade Utilities diskette.

4.6.1 mp_prep Shell Script

This shell script prepares the UP system for the migration. It installs filesets that are required to enable the MP kernel and support SMP specific devices. It also create the links that are necessary to build an MP boot image.

```
#!/usr/bin/ksh
#
#   Function: mp_prep
#
#   Description: Prepares a system for hardware upgrade from UP to MP
#
#   ORIGINS: 27
#
#   (C) COPYRIGHT International Business Machines Corp. 1995
#   All Rights Reserved
#   Licensed Materials - Property of IBM
#   US Government Users Restricted Rights - Use, duplication or
#   disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#

# Make sure that this is being executed with root authority
check_root () {
    idname= id
    if [ 0 -ne  expr "$idname" : ".*root"  ]
       then return 0
       else {
        echo "You must have root permission to successfully run this program."
        exit 1
       }
    fi
}

# Make sure that the system is at least at 4.1.2
check_412 () {

    [[ $1 -gt 4 ]] && return 0
    [[ $1 -lt 4 ]] && return 1

    [[ $2 -gt 1 ]] && return 0
    [[ $2 -lt 1 ]] && return 1
```

```
                    [[ $3 -ge 2 ]] && return 0

                    #
                    # List the maintenance levels which have been seen by this system

                    echo
                    echo
                    echo "ERROR: This system is not fully installed with AIX 4.1.2 or greater."
                    echo "       The highest completely installed level is $1.$2.$3.$4"
                    echo
                    echo "The levels of AIX known to this system are:"
                    /usr/bin/oslevel -q 2>/dev/null
                    echo
                    echo "If the only level of AIX known to this system level is 4.1.0.0,"
                    echo "then the system must be completely reinstalled with AIX 4.1.2.0 or later."

                    return 1
            }

            # Introduction
            introduction () {
                    echo
                    echo "This script performs the software operations required to install"
                    echo "the multi-processor support necessary to prepare a system backup"
                    echo "image to be used after converting a uni-processor system to a multi-processor"
                    echo "system."
                    echo
                    echo "Continue? (y/n) \c"

                    read ans
                    [[ -n "$ans" ]] && [[ "$ans" != "y" ]] && {
                        echo "Exiting at user request"
                      exit 8
                    }
            }

            # Installation device used as input
            if [[ $# > 1 ]] || [[ $1 = -* ]]
            then
                    echo "Usage: mp_prep <installation device>"

                    check_root
                    if [ $? -eq 0 ]
                    then
                        echo
                        echo "Known installation devices are listed below:"
                     /usr/lib/instl/sm_inst list_devices
                    fi
                    exit 1
            fi

            nparams=$#
            inputdev=$1

            introduction

            check_root
```

```
version=$(/usr/bin/uname -v)
[[ $version -lt 4 ]] && {
        echo
        echo
        echo "ERROR: The system must be installed with at least AIX 4.1.2"
        echo "        This system is installed with some form of Version $version."
        exit 3
}

# Call oslevel to make sure that this system is at least at 4.1.2.0
echo "Verifying that system is installed with at least AIX 4.1.2...\c"
curlevel=$(/usr/bin/oslevel)
[[ -z "$curlevel" ]] && {
    echo
    echo
    echo "ERROR determining system level - could not run /usr/bin/oslevel"
    exit 2
}

# Determine if we're at least at 4.1.2.0
OIFS="$IFS"
IFS="."
set -- $curlevel
check_412 $1 $2 $3 $4

[[ $? -ne 0 ]] && exit 3
# Finish status of 412 check
echo ok

IFS="$OIFS"

if [[ $nparams -eq 0 ]]
then
    echo "Known installation devices are listed below:"
    /usr/lib/instl/sm_inst list_devices
    echo
    while [[ -z "$DEVICE" ]]
    do
        echo "Enter the name of the installation device to be used: \c"
        read DEVICE < /dev/tty
    done
else
    DEVICE=$inputdev
fi


# Call sm_inst to handle the device manipulation and the installation

# Get system language from /var/adm/ras/bosinst.data
mpmsg=
SYSLANG= grep -v "¬\#" /var/adm/ras/bosinst.data | grep "MESSAGES =" | awk '{print $3}'

[[ -n "$SYSLANG" ]]  && [[ "$SYSLANG" != "C" ]] && {
    mpmsg="bos.msg.$SYSLANG.rte.mp"
}
```

```
mp_filesets="bos.rte.mp bos.sysmgt.sysbr devices.mca.fed9 devices.mca.8efc \
devices.rs6ksmp.base $mpmsg"

echo "Installing multi-processor and system backup software support..."
/usr/lib/instl/sm_inst installp_cmd -T iems -q -a -X -d $DEVICE -e /tmp/mp_prep.log \
$mp_filesets

[[ $? -ne 0 ]] && {
    echo
    echo "ERROR: Failed to install multi-processor software support."
    [[ -f /tmp/mp_prep.log ]] && {
        echo "        See /tmp/mp_prep.log for further information."
    }
    exit 4
}

# Link in the mp kernel
[[ -f /usr/lib/boot/unix_mp ]] || {
    echo
    echo "ERROR: /usr/lib/boot/unix_mp does not exist."
    [[ -f /tmp/mp_prep.log ]] && {
        echo "        See /tmp/mp_prep.log for further information."
    }

    exit 5
}

/usr/bin/rm -f /unix /usr/lib/boot/unix

/usr/bin/ln -s /usr/lib/boot/unix_mp /unix || {
    echo
    echo "ERROR: Failure linking /usr/lib/boot/unix_mp to /unix."
    exit 6
}

/usr/bin/ln -s /usr/lib/boot/unix_mp /usr/lib/boot/unix || {
    echo "ERROR: Failure linking /usr/lib/boot/unix_mp to /usr/lib/boot/unix."
    exit 7
}

# Fix the .proto files if necessary so that the tape can boot mp and up

[[ ! -f /usr/lib/boot/protoext/tape.proto.ext.base.com.upmp ]] && {
  cat >/usr/lib/boot/protoext/tape.proto.ext.base.com.upmp <<EOF
cfgsys      ---- 777 0 0 /usr/lib/methods/cfgsys
cfgsys_p    ---- 777 0 0 /usr/lib/methods/cfgsys_p
EOF
}

/usr/bin/cat /tmp/mp_prep.log >> /smit.log 2>/dev/null
[ $? -eq 0 ] && {
    echo "The software installation log has been appended to /smit.log"
    echo "for future reference."
}
/usr/bin/rm -f /tmp/mp_prep.log

echo
echo "System is now ready to be backed up in preparation for the hardware upgrade."
```

```
        echo "Insert blank tape cartridge into the appropriate tape drive and run"
        echo "\"smitty mksysb\" to complete preparation."
        echo

        exit 0
```

4.6.2 mach.upg Shell Script

This shell script removes some UP device definitions from the ODM that do not exist on the SMP and some device definitions that need to be rebuilt on the SMP after the mksysb is restored.

```
#!/bin/ksh
#
# COMPONENT_NAME: (CFGMETHODS) Configuration Methods
#
# FUNCTIONS: mach.upg
#
# ORIGINS: 27
#
# (C) COPYRIGHT International Business Machines Corp. 1992,1995
# All Rights Reserved
# Licensed Materials - Property of IBM
#
# US Government Users Restricted Rights - Use, duplication or
# disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#**********************************************************************
#
#  NAME    : mach.upg (machine upgrade)
#  PURPOSE : This script prepares a machine for a hardware upgrade.
#            It should be run BEFORE shutting down the machine
#            to perform the upgrade.
#
#--------------------------------------------------------------------
#
#  WARNING: This script will remove items from the ODM database!!
#
#**********************************************************************


function disk_msg
{
    echo "\n\n"
    echo "You must specify the name of the boot disk!"
    echo "usage: $ME boot_disk_name"
    echo "\n\n"
    echo "***** WARNING!!"
    echo "***** WARNING!! The boot disk name may be different in a"
    echo "***** WARNING!! diskette or tape boot than it is in the disk"
    echo "***** WARNING!! based boot. Double check the location codes"
    echo "***** WARNING!! in the list below and use the name associated"
    echo "***** WARNING!! with the correct location code!"
    echo "*****"
    echo "\nCurrent disk configuration is:"
    ODMDIR=/dev/objrepos
    lsdev -C -cdisk
```

```
                ODMDIR=/etc/objrepos
        }

        function delete_dev_list
        {
            for dvc in $DEL_LIST
            do
                echo "Removing $dvc . . ."
                odmdelete -q name=$dvc -o CuDv
                odmdelete -q name=$dvc -o CuAt
                odmdelete -q name=$dvc -o CuVPD
            done
        }

        function warn_msg
        {
            echo "\n\n"
            echo "***** WARNING!! $WARNMSG !!"
            read INPUT?"Do you wish to continue? (y/n) [n]:"
            case $INPUT in
                y* | Y*)
                    ;;
                *)
                    exit 0
                    ;;
            esac
        }



        # Start of script

        ME= basename $0
        BOOTDEV= /usr/sbin/bootinfo -b

        if [[ "$#" < "1" ]]
        then
            case $BOOTDEV in
                hdisk*)
                    DISK=$BOOTDEV
                    ;;
                *)
                    disk_msg
                    exit -1
                    ;;
            esac

        else
            DISK= basename $1
            if [[ $DISK != $BOOTDEV ]]
            then
                case $BOOTDEV in
                    hdisk*)
                        DISK=$BOOTDEV
                        ;;
                    *)
                        x= lsdev -C -l $DISK -S AVAILABLE -F uniquetype
                        if [[ "$x" = "" ]]
```

```
                then
                    echo "\n\n**ERROR: The specified disk ($1) is not
                    currently available."
                    disk_msg
                    exit -1
                fi
                ;;
            esac
        fi
fi
echo "\nBoot disk is: $BOOTDEV"

WARNMSG="ODM data will be modified"
warn_msg

echo "\n\n"
read INPUT?"Is this for a Uniprocessor to Multiprocessor upgrade?
(y/n) [n]:"
case $INPUT in
    y* | Y*)
        WARNMSG="Boot image will be rebuilt"
        warn_msg
        echo "\n\nRelinking /unix to /unix_mp"
        rm /unix
        ln -s /usr/lib/boot/unix_mp /unix
        cd /usr/lib/boot
        rm unix
        ln -s /usr/lib/boot/unix_mp unix
        echo "Rebuilding MP boot image.\n\n"
        echo "Please wait while bosboot is being processed.\n\n"
        bosboot -ad /dev/$BOOTDEV -k /usr/lib/boot/unix_mp -T rs6ksmp
        ;;
    *)
        ;;
esac


# First delete all standard I/O adapters
DEL_LIST= lsdev -C -s sio -F name
delete_dev_list


# Now delete the sio0 device
DEL_LIST= lsdev -C -l sio0 -F name
delete_dev_list


# Next delete all of the devices related to the serial optical link
DEL_LIST= lsdev -C -t ops -F name
delete_dev_list
DEL_LIST= lsdev -C -s otp -F name
delete_dev_list
DEL_LIST= lsdev -C -s slc -F name
delete_dev_list


# Next, delete all of the "system" devices like ioplanar0,
# sysplanar0, serial optical link chips, memory, etc.
```

```
DEL_LIST= lsdev -C -s sys -F name
delete_dev_list

# Finally, delete the sys0 device
DEL_LIST= lsdev -C -c sys -F name
delete_dev_list


customization
echo "Saving base customization\n"
savebase

exit 0
```

4.7 Unsupported Hardware on the IBM SMP Models

This is a list of features that are currently not supported on the SMP systems and cannot be moved across during a UP to SMP upgrade. Some of these features might be supported on the SMP systems in the future. Please consult IBM announcement letters to check which hardware is currently not supported on the SMP systems at the time of the upgrade.

4.7.1 Model G30

The following features are currently not supported on the G30 SMP system.

Feature   Description
- 1101 Non-Locking Switch Select
- 1906 Fiber Channel Adapter/266
- 2121 160 MB SCSI Disk Drive
- 2400 M-Video Capture Adapter (NTSC)
- 2401 M-Video Capture Adapter (PAL)
- 2402 IBM Network Terminal Accelerator - 256 Session
- 2403 IBM Network Terminal Accelerator - 2048 Session
- 2404 IBM Ultimedia Video I/O Adapter
- 2405 IBM Ultimedia Video Compression/Decompression Card
- 2560 400 MB SCSI Disk Drive
- 2563 160 MB to 400 MB SCSI Disk Drive Select
- 2711 POWER Gt4xi 8-bit Graphics Adapter
- 2712 POWER Gt4xi 24-bit Graphics Adapter
- 2713 POWER Gt4i 24-bit Graphics Adapter
- 2754 S/390 ESCON Channel Emulator
- 2759 System/370 Channel Emulator/A
- 2760 Grayscale Graphics Display Adapter
- 2768 POWER Gt3i Graphics Adapter
- 2770 Color Graphics Display Adapter
- 2776 POWER Gt4e Graphics Adapter
- 2777 POWER Gt3 Graphics Adapter
- 2780 High-Performance 8-bit 3D Color Graphics Processor
- 2781 High-Performance 24-bit 3D Color Graphics Processor
- 2782 24-bit Z-Buffer Solid Rendering Option
- 2783 24-bit Color Graphics Frame Buffer Upgrade
- 2790 POWER Gt4x 8-bit Feature
- 2791 POWER Gt4x 24-bit Feature
- 2792 POWER Gt4 8-bit to 24-bit Upgrade

- 2794 POWER Gt4 Performance Upgrade
- 2795 POWER Gt4 8-bit Feature
- 2796 POWER Gt4 24-bit Feature
- 2800 S/370 Host Interface Adapter
- 2801 5086 Attachement Adapter
- 2802 5085 Attachement Adapter
- 2810 Graphics Input Device Adapter
- 2811 Graphics Input Device Cable
- 2833 Integrated SCSI Controller Cable
- 2922 ARTIC960 Cable 4-Port EIA/232
- 2972 IBM Auto Token-Ring LANstreamer 32 MC Adapter
- 2992 Ethernet/FDX 10 Mbps TP/AUI MC Adapter
- 4040 0.5 MB L2 Cache
- 4041 1 MB L2 Cache
- 4063 8 MB HD3 Memory Card
- 4066 16 MB HD3 Memory Card
- 4067 32 MB HD3 Memory Card
- 4068 16 MB to 32 MB HD3 Memory Select
- 4069 64 MB HD3 Memory Card
- 4071 32 MB to 64 MB HD3 Memory Select
- 4074 16 MB to 64 MB HD3 Memory Select
- 4090 128 MB Memory Card
- 4091 16 MB to 128 MB Memory Select
- 4092 32 MB to 128 MB Memory Select
- 4095 256 MB Memory Card
- 4096 32 MB to 256 MB Memory Select
- 4099 16 MB to 256 MB Memory Select
- 4221 Ethernet AUI/Thin Riser
- 4222 Ethernet Twisted Pair Riser
- 4350 POWERgraphics GTO Accelerator Feature
- 5032 32 MB Memory SIMM Kit
- 5064 64 MB Memory SIMM Kit
- 5128 128 MB Memory SIMM Kit
- 6214 High Performance 4-Port SSA Adapter
- 6301 M-Audio Capture Playback Adapter
- 6302 IBM Ultimedia Audio Adapter
- 6306 IBM Speech Accelerator 1
- 6307 IBM Speech Accelerator 2
- 6400 64-Port Asynchronous Controller
- 6401 16-Port Asynchronous Concentrator
- 6402 RJ-45 to DB-25 Converter Cable
- 6509 Media Bay Disk Drive Mounting Kit
- 6531 19-Inch Equipment Rack Mounting Hardware
- 6532 Rack Mount/Environmentally-Hardened Option
- 8A0836 MG24 Graphics Adapter
- 9000 Ethernet AUI/Thin Riser Specify
- 9001 Ethernet Twisted Pair Riser Specify
- 9115 Transformer Specify, 100 to 110 VAC
- 9116 Transformer Specify, 115 to 127 VAC
- 9117 Transformer Specify, 200 to 220 VAC
- 9118 Transformer Specify, 230 to 240 VAC
- 9234 16 MB HD3 Memory Specify
- 9235 32 MB HD3 Memory Specify
- 9242 160 MB SCSI Disk Drive Specify

- 9244 400 MB SCSI Disk Drive Specify
- 9892 Power Cord Specify - Linecord to PDU

4.7.2 Model J30

The following features are currently not supported on the J30 deskside SMP system.

Feature   Description
- 1101 Non-Locking Switch Select
- 1110 Keyboard - US English Soft Touch (US/CANADA Only)
- 1902 Optics Daughter Card, 1-Port 1063 Mbps
- 1904 Fibre Channel Adapter/1063
- 1906 Fiber Channel Adapter/266
- 2400 M-Video Capture Adapter (NTSC)
- 2401 M-Video Capture Adapter (PAL) (AP, EMEA)
- 2402 IBM Network Terminal Accelerator - 256 Session
- 2403 IBM Network Terminal Accelerator - 2048 Session
- 2404 IBM Ultimedia Video I/O Adapter
- 2405 IBM Ultimedia Video Compression/Decompression Card
- 2428 8-bit Internal 6-Drop Cable for IBM SCSI-2 Fast/Wide Adapters Select
- 2429 16-bit Internal 6-Drop Cable for IBM SCSI-2 Fast/Wide Adapters Select
- 2430 8-bit Internal 6-Drop Cable for IBM SCSI-2 Fast/Wide Adapters
- 2431 16-bit Internal 6-Drop Cable for IBM SCSI-2 Fast/Wide Adapters
- 2433 16-bit Internal 6-Drop Cable for IBM SCSI-2 Fast/Wide Adapters Select
- 2500 355 MB SCSI Disk Drive
- 2510 670 MB SCSI Disk Drive
- 2511 355 MB to 670 MB SCSI Disk Drive Select Feature
- 2529 400 MB to 857 MB SCSI Disk Drive Select
- 2530 857 MB SCSI Disk Drive
- 2531 355 MB to 857 MB SCSI Disk Drive Select
- 2532 640 MB to 857 MB SCSI Disk Drive Select
- 2534 800 MB to 857 MB SCSI Disk Drive Select
- 2542 640 MB SCSI Disk Drive Pair
- 2543 355 MB to 640 MB SCSI Disk Drive Pair Select
- 2560 400 MB SCSI Disk Drive
- 2562 800 MB SCSI Disk Drive Pair
- 2570 1.37 GB SCSI Disk Drive
- 2572 400 MB to 1.37 GB SCSI Disk Drive Select
- 2574 800 MB to 1.37 GB SCSI Disk Drive Select
- 2578 1.37 GB SCSI Disk Drive (EMEA PRICING FEATURE)
- 2590 2.4 GB SCSI-2 Disk Drive
- 2592 400 MB to 2.4 GB SCSI-2 Disk Drive Select
- 2593 670 MB to 2.4 GB SCSI-2 Disk Drive Select
- 2594 2.0 GB to 2.4 GB SCSI-2 Disk Drive Select
- 2596 2.4 GB SCSI-2 Disk Drive (EMEA PRICING FEATURE)
- 2598 2 GB to 2.4 GB SCSI-2 Disk Drive Select
- 2615 External 5 1/4-Inch Diskette Drive Attachment Cable
- 2650 POWER GXT150M Graphics Adapter
- 2711 POWER Gt4xi 8-bit Graphics Adapter
- 2712 POWER Gt4xi 24-bit Graphics Adapter
- 2713 POWER Gt4i 24-bit Graphics Adapter
- 2735 High-Performance Parallel Interface Adapter
- 2754 S/390 ESCON Channel Emulator
- 2759 System/370 Channel Emulator/A

- 2760 Grayscale Graphics Display Adapter
- 2768 POWER Gt3i Graphics Adapter
- 2770 Color Graphics Display Adapter
- 2776 POWER Gt4e Graphics Adapter
- 2777 POWER Gt3 Graphics Adapter
- 2780 High-Performance 8-bit 3D Color Graphics Processor
- 2781 High-Performance 24-bit 3D Color Graphics Processor
- 2782 24-bit Z-Buffer Solid Rendering Option
- 2783 24-bit Color Graphics Frame Buffer Upgrade
- 2790 POWER Gt4x 8-bit Feature
- 2791 POWER Gt4x 24-bit Feature
- 2792 POWER Gt4 8-bit to 24-bit Upgrade
- 2794 POWER Gt4 Performance Upgrade
- 2795 POWER Gt4 8-bit Feature
- 2796 POWER Gt4 24-bit Feature
- 2810 Graphics Input Device Adapter
- 2811 Graphics Input Device Cable
- 2820 IBM 7250 Attachment Adapter
- 2829 External I/O Controller - Towers
- 2832 SCSI Controller Cable
- 2833 Integrated SCSI Controller Cable
- 2860 Serial Optical Channel Converter
- 2866 6-Meter Serial Optical Channel Converter Cable
- 2867 10-Meter Serial Optical Channel Converter Cable
- 2868 20-Meter Serial Optical Channel Converter Cable
- 2869 60-Meter Serial Optical Channel Converter Cable
- 2870 100-Meter Serial Optical Channel Converter Cable
- 2922 Cable Option EIA 232
- 2972 IBM Auto Token-Ring LANstreamer 32 MC Adapter
- 2992 Ethernet/FDX 10 Mbps TP/AUI MC Adapter
- 3600 POWERdisplay 16
- 3601 POWERdisplay 19
- 3607 POWERdisplay 17
- 3608 POWERdisplay 20
- 3612 P50 Color Monitor
- 3613 P70 Color Monitor
- 3614 P200 Color Monitor
- 3615 P201 Color Monitor
- 4008 8 MB SD1 Memory Card
- 4010 8 MB to 16 MB SD1 Memory Select Feature
- 4016 16 MB SD1 Memory Card
- 4032 32 MB HD1 Memory Card
- 4033 8 MB to 32 MB HD1 Memory Select
- 4035 64 MB HD1 Memory Card
- 4036 8 MB to 64 MB HD1 Memory Select
- 4063 8 MB HD3 Memory Card
- 4065 32 MB HD2 Memory Card
- 4066 16 MB HD3 Memory Card
- 4067 32 MB HD3 Memory Card
- 4068 16 MB to 32 MB HD3 Memory Select
- 4069 64 MB HD3 Memory Card
- 4070 16 MB to 64 MB HD3 Memory Select
- 4071 32 MB to 64 MB HD3 Memory Select
- 4090 128 MB Memory Card

- 4092 32 MB to 128 MB Memory Select
- 4095 256 MB Memory Card
- 4096 32 MB to 256 MB Memory Select
- 4213 13W3 to 15-Pin D-Shell Converter Cable
- 4214 13W3 to 60/77Hz Display Cable
- 4221 Ethernet AUI/Thin Riser
- 4222 Ethernet Twisted Pair Riser
- 4227 Sun Compatible Display Converter Cable
- 4229 13W3 to PowerDisplay 16S Display Cable
- 4234 13W3 to 13W3 Display Cable
- 4350 POWERgraphics GTO Accelerator Feature
- 5032 32 MB Memory SIMM Kit
- 5064 64 MB Memory SIMM Kit
- 5128 128 MB Memory SIMM Kit
- 6010 Keyboard - 101 Keys (US)
- 6011 Keyboard - 102 Keys (Belgian-Dutch/French)
- 6012 Keyboard - 102 Keys (Canadian French)
- 6013 Keyboard - 102 Keys (Danish)
- 6014 Keyboard - 102 Keys (Finnish)
- 6015 Keyboard - 102 Keys (French)
- 6016 Keyboard - 102 Keys (German)
- 6017 Keyboard - 102 Keys (Italian)
- 6018 Keyboard - 102 Keys (Brazilian)
- 6019 Keyboard - 102 Keys (Norwegian)
- 6020 Keyboard - 102 Keys (Portuguese)
- 6021 Keyboard - 102 Keys (Spanish)
- 6022 Keyboard - 102 Keys (Swiss)
- 6023 Keyboard - 102 Keys (United Kingdom - English)
- 6024 Keyboard - 102 Keys (Icelandic)
- 6025 Keyboard - 102 Keys (Turkish)
- 6026 Keyboard - 102 Keys (Greek)
- 6027 Keyboard - 102 Keys (Hebrew)
- 6028 Keyboard - 102 Keys (Arabic)
- 6030 Keyboard - 106 Keys (Japanese - Kanji)
- 6031 Keyboard - 106 Keys (Korean)
- 6033 Keyboard - 106 Keys (Chinese Traditional - Taiwan)
- 6034 Keyboard - Dutch #143 (Netherlands)
- 6035 Keyboard - Turkish #440
- 6041 3-Button Mouse
- 6143 CD-ROM to 2.3 GB 8MM Internal Tape Drive Select
- 6146 2.3 GB 8MM Internal Tape Drive
- 6301 M-Audio Capture Playback Adapter
- 6306 IBM Speech Accelerator 1
- 6307 IBM Speech Accelerator 2
- 6400 64-Port Asynchronous Controller
- 6401 16-Port Asynchronous Concentrator
- 6402 RJ-45 to DB-25 Converter Cable
- 6501 3 1/2-Inch Disk Drive Mounting Hardware
- 6506 Incremental Cooling Fan
- 8A083 MG24 Graphics Adapter
- 8A0875 ICL Search Accelerator
- 9000 Ethernet AUI/Thin Riser Specify
- 9001 Ethernet Twisted Pair Riser Specify
- 9042 1 MB L2 Cache Specify

- 9115 Transformer Specify, 100 to 110 VAC
- 9116 Transformer Specify, 115 to 127 VAC
- 9117 Transformer Specify, 200 to 220 VAC
- 9118 Transformer Specify, 230 to 240 VAC
- 9150 Serial Optical Channel Converter Module Specify
- 9220 SCSI I/O Controller Specify
- 9231 8 MB SD1 Memory Specify
- 9232 32 MB HD2 Memory Specify
- 9234 16 MB HD3 Memory Specify
- 9235 32 MB HD3 Memory Specify
- 9236 16 MB SD1 Memory Specify
- 9243 640 MB SCSI Disk Drive Specify
- 9244 400 MB SCSI Disk Drive Specify
- 9245 800 MB SCSI Disk Drive Specify
- 9246 355 MB SCSI Disk Drive Specify
- 9430 SCSI-2 Fast/Wide Internal Cable Specify

4.7.3 Model R3U

The following features are currently not supported on the model R3U rack SMP.

Feature   Description
- 1902 Optics Daughter Card, 1-Port 1063 Mbps
- 1904 Fibre Channel Adapter/1063
- 1906 Fiber Channel Adapter/266
- 2402 IBM Network Terminal Accelerator - 256 Session
- 2403 IBM Network Terminal Accelerator - 2048 Session
- 2620 Internal 5 1/4-Inch Diskette Drive
- 2735 High-Performance Parallel Interface Adapter
- 2754 S/390 ESCON Channel Emulator
- 2759 System/370 Channel Emulator/A
- 2800 System/370 Host Interface Adapter
- 2801 5086 Attachment Adapter
- 2802 5085 Attachment Adapter
- 2832 SCSI Controller Cable
- 2835 SCSI High-Performance External I/O Controller
- 2860 Serial Optical Channel Converter
- 2866 6-Meter Serial Optical Channel Converter Cable
- 2867 6-Meter Serial Optical Channel Converter Cable
- 2868 20-Meter Serial Optical Channel Converter Cable
- 2869 60-Meter Serial Optical Channel Converter Cable
- 2870 100-Meter Serial Optical Channel Converter Cable
- 2922 Cable Option EIA 232
- 2972 IBM Auto Token-Ring LANstreamer 32 MC Adapter
- 2992 Ethernet/FDX 10 Mbps TP/AUI MC Adapter
- 3120 External SCSI Controller Cable (60-Pin Connector)
- 3122 Serial to Re-IPL Port Cable for Drawer/Drawer
- 3123 Serial to Re-IPL Port Cable for Rack/Rack
- 4008 8 MB SD1 Memory Card
- 4010 8 MB to 16MB SD1 Memory Select Feature
- 4016 16 MB SD1 Memory Card
- 4032 32 MB HD1 Memory Card
- 4033 8 MB to 32 MB HD1 Memory Select
- 4035 64 MB HD1 Memory Card
- 4036 8 MB to 64 MB HD1 Memory Select

- 4063 8 MB HD3 Memory Card
- 4066 16 MB HD3 Memory Card
- 4067 32 MB HD3 Memory Card
- 4068 16 MB to 32 MB HD3 Memory Select
- 4069 64 MB HD3 Memory Card
- 4070 16 MB to 64 MB HD3 Memory Select
- 4071 32 MB to 64 MB HD3 Memory Select
- 4092 32 MB to 128 MB Memory Select
- 4090 128 MB Memory Card
- 4094 64 MB to 128 MB Memory Select
- 4095 256 MB Memory Card
- 4097 64 MB to 256 MB Memory Select
- 5032 32 MB Memory SIMM Kit
- 5064 64 MB Memory SIMM Kit
- 5128 128 MB Memory SIMM Kit
- 6002 Asynchronous Expansion Drawer
- 6003 Micro Channel Expansion Board (8-slot)
- 6229 DC Input CPU Enclosure
- 6301 M-Audio Capture Playback Adapter
- 6307 IBM Speech Accelerator 2
- 6400 64-Port Asynchronous Controller
- 6401 16-Port Asynchronous Concentrator
- 6402 RJ-45 to DB-25 Converter Cable
- 6894 Remote Reset Port
- 9003 Micro Channel Expansion Board Specify
- 9043 1.0 MB Base L2 Cache Card (10ns-pairs req'd)
- 9115 Transformer Specify, 100 to 110 VAC
- 9116 Transformer Specify, 115 to 127 VAC
- 9117 Transformer Specify, 200 to 220 VAC
- 9118 Transformer Specify, 230 to 240 VAC
- 9150 Serial Optical Channel Converter Module Specify
- 9220 SCSI I/O Controller Specify
- 9226 CPU Drawer Specify
- 9227 CPU Drawer Specify for AC-Models
- 9231 8 MB SD1 Memory Specify
- 9234 16 MB HD3 Memory Specify
- 9235 32 MB HD3 Memory Specify
- 9236 16 MB SD1 Memory Specify
- 9238 64 MB HD3 Memory Specify

4.7.4 Model R30

The following features are currently not supported on the model R30 rack SMP.

Feature   Description
- 1902 Optics Daughter Card, 1-Port 1063 Mbps
- 1904 Fibre Channel Adapter/1063
- 1906 Fiber Channel Adapter/266
- 2402 IBM Network Terminal Accelerator - 256 Session
- 2403 IBM Network Terminal Accelerator - 2048 Session
- 2586 2 GB SCSI-2 Fast/Wide Disk Drive
- 2735 High-Performance Parallel Interface Adapter
- 2754 S/390 ESCON Channel Emulator
- 2759 System/370 Channel Emulator/A
- 2800 System/370 Host Interface Adapter

- 2832　SCSI Controller Cable
- 2835　SCSI High-Performance External I/O Controller
- 2922　Cable Option EIA 232
- 2972　IBM Auto Token-Ring LANstreamer 32 MC Adapter
- 2992　Ethernet/FDX 10 Mbps TP/AUI MC Adapter
- 3032　1.1 GB SCSI-2 Fast/Wide Disk Drive
- 3033　2.2 GB SCSI-2 Fast/Wide Disk Drive
- 3034　4.5 GB SCSI-2 Fast/Wide Disk Drive
- 3042　1.1 GB SCSI-2 Fast/Wide Disk Drive Select (Must order quantity of two)
- 3045　2.2 GB SCSI-2 Fast/Wide Disk Drive Select
- 3049　4.5 GB SCSI-2 Fast/Wide Disk Drive Select
- 3064　2 GB SCSI-2 Fast/Wide Initial Order (Must order quantity of two)
- 3120　External SCSI Controller Cable (60-Pin Connector)
- 3122　Serial to Re-IPL Port Cable for Drawer/Drawer
- 3123　Serial to Re-IPL Port Cable for Rack/Rack
- 4067　32 MB HD3 Memory Card
- 4069　64 MB HD3 Memory Card
- 4090　128 MB Memory Card
- 4094　64 MB to 128 MB Memory Select
- 4095　256 MB Memory Card
- 4097　64 MB to 256 MB Memory Select
- 5064　64 MB Memory SIMM Kit
- 5128　128 MB Memory SIMM Kit
- 6229　DC Input CPU Enclosure
- 6301　M-Audio Capture Playback Adapter
- 6894　Remote Reset Port
- 9003　Micro Channel Expansion Board Specify
- 9042　1 MB L2 Cache Specify
- 9043　1.0 MB BASE L2 CACHE CARD (10NS-PAIRS REQ'D)
- 9227　CPU Drawer Specify for AC-Models
- 9238　64 MB HD3 Memory Specify
- 9264　2.0 GB Base Fast/Wide Disk Drive
- 9290　CPU Drawer Specify
- 9893　Drawer Glide and Cable Management Hardware Specify
- 9894　Remote Reset Port Specify

Chapter 5. Migrating from AIX V3.2.5 to AIX V4.1 with Oracle

This chapter covers the migration of an Oracle environment from AIX V3.2.5 to AIX V4.1.3. The objective is to give you some guidelines, some hints and tips that might be useful to achieve the migration of an entire Oracle environment.

5.1 Related Publications

We recommend that you have the following documents for reference during the planning and the migration.

- *Oracle7 Server Migration Guide*

- *Oracle7 for AIX-Based Systems Installation & Configuration Guide*

- *SQL*Net Version 2.0 Administrator's Guide*

See also "Related Publications" on page xxi for publications related to AIX.

5.2 Introduction

This section introduces basic concepts of the Oracle Relational Database Management System (RDBMS). If you are familiar with Oracle you can skip this section and go to the planning section. We discuss the Oracle structure and some concepts that are referred to later in this chapter. We see how Oracle files are organized in AIX. This information is used to develop a backup/recovery plan.

5.2.1 Oracle Files Organization

Oracle files in AIX can be set into three main categories according to their functionality:

1. Oracle code itself:

   Oracle code is made up of executable programs, libraries, message files, and SQL scripts that come with the Oracle distribution media. These files are only updated if you install a fix (or patch) or a new release.

   During installation, you can specify the directory in which to load Oracle code. This directory is called ORACLE_HOME. All of the Oracle programs and files are located under the ORACLE_HOME directory, except for some configuration files that will be created under /etc and /usr.

2. Database:

   A database contains user data and control information used by Oracle. Each database has its own unique name. The database name is called its system identifier or SID.

   The database is composed of:

   - Configuration files: These files contain parameters used to start up the database and tuning parameters, for example the buffer size.

   - Control files: These files contain control information used by Oracle. They contain information about other data files and log files. These files are very important. Usually, it is recommended to have more than one copy of the control files on different physical disks to guard against a disk failure.

- Data files: These files contain user data (tables) stored in the database. They also contain indexes, temporary tables, rollback segments, and the Oracle system catalog.

- Log files: These files (sometimes called redo log files) contain transaction log records. They record all the changes that have been made to the database. If a machine is stopped without a database shutdown, then when the database is restarted, the database will use the information stored in the log for recovery. The database will be brought back up to the last committed transaction.

Figure 48 shows the relationship between these files for one database.

3. Application files:

Application files are files that contain the source code, object code and executable programs of an application. In this section, application stands for all the programs created by a user or a third party vendor, such as forms created by SQL*Forms or Reports created by SQL*Report.
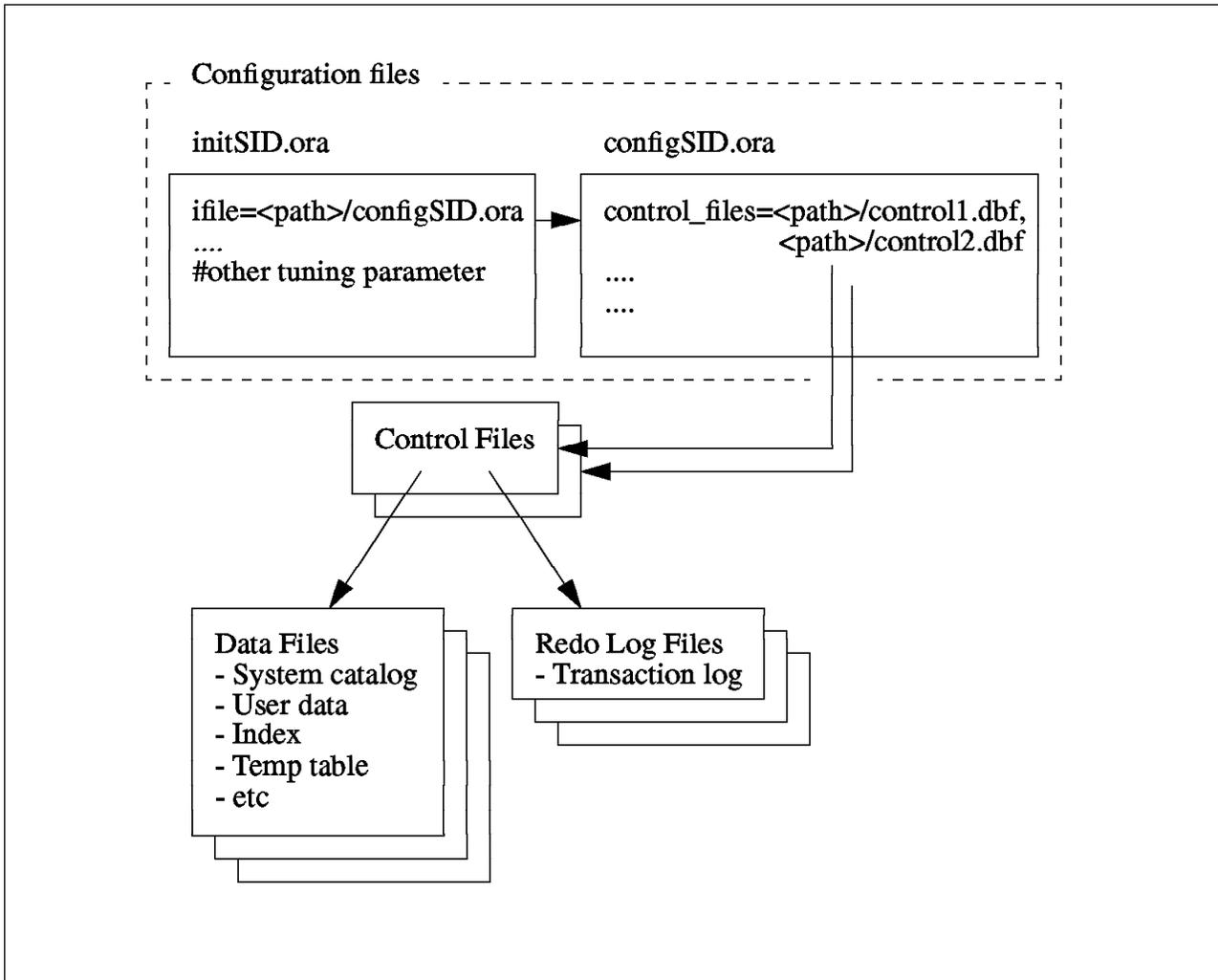
Figure 48. Oracle Database Files Organization

5.2.2 Raw Devices

Most database vendors, including Oracle, try to reduce disk I/O overhead by bypassing UNIX file services communicating directly with disks device drivers. The data is stored in a hard disk partition (or a logical volume in AIX) instead of a normal UNIX file in a filesystem. In cases where the database server reads/writes data from/to the device directly, these operations bypass a normal UNIX buffer cache and use less time in operating system code. A logical volume that contains the database is called a *raw device.* Strictly speaking, UNIX supports raw devices other than disk, but in this chapter the terminology raw device will be used to refer to a raw disk device or raw Logical Volume (LV) device. The database administrator can choose to store the database files in either raw devices or in normal UNIX files.

In AIX, you can create a raw device by creating a logical volume. For example, if you create a logical volume, `lv01`, AIX will create a character device, `/dev/rlv01`, in the /dev directory. Then you can configure the Oracle database to use `/dev/rlv01` for storing a data or log file.

5.2.3 Oracle-Related Users and Groups

Oracle uses standard AIX security features to identify the following special users and groups. These users are responsible for administering the database, installing software and maintaining the database.

Note: Oracle also uses its own user and password authentication. The Oracle userids and related encripted passwords are stored in the ORACLE table DBA_USERS and are preserved during the migration.

You must have access to the following user accounts in order to perform an Oracle software upgrade:

- root user:

    A root user is involved in the pre-installation process. Oracle needs root authority to set up the environment, for example, to configure a device, load kernel extensions and so on.

- Oracle owner:

    The Oracle owner is a user who has the highest privilege in the database. The owner can do everything to the database, such as installing a new software or a patch, creating a database and so on. Usually, the Oracle owner's name is *oracle*, but it is not mandatory. You can give the Oracle owner any name, or you may have several owners for different releases of Oracle. Please check with your database administrator to identify the Oracle owner's name in your system. However, during the migration, we recommend NOT changing the Oracle owner's name in order to maintain ownership of the existing files.

- dba group:

    The database administrator group, or dba group, is a group that has privileges to perform administrative tasks in the database. A user who is a member of this group can access SQL*DBA, without giving a password, to manage the database (start/stop the database, create/drop data files and so on). The Oracle owner must be a member of this group. Usually, membership of this group is only granted to few users.

The Oracle owner's name and the dba group name will be asked during the Oracle software installation. Then, ownership of the database files will be changed to the user and group that you have specified.

5.2.4 Oracle Install Process

The purpose of this section is to give you some ideas about the basic steps in the Oracle installation. You can find more information in the *Oracle for AIX Installation & Configuration Guide*. Most of the installation process is performed by the Oracle owner, for some steps that need root privileges.

In order to install Oracle, you need to do the following:

- Restore the Oracle Installer:

  In Oracle V7, you have to restore the Installer program from the distribution media to disk. The Installer has a menu interface. This interface will help the database administrator install the rest of the Oracle software.

- Run the pre-installation script:

  Oracle requires a specific configuration in AIX and provides a shell script, called rootpre.sh, to reconfigure the system. This script must be executed by root. It makes two changes to the AIX configuration:

  – It configures an asynchronous I/O device:

    Oracle uses asynchronous I/O to access data files stored on disks. So, the device, **aio0**, in AIX must be available. The rootpre.sh script will change the aio0 device parameters so that it becomes available on every system restart.

  – It loads kernel extensions:

    The database manager is responsible for handling resources contention. Traditional UNIX uses semaphores to suspend and wake up a process when the required resource is unavailable. Oracle has implemented a post/wait algorithm to use instead of semaphores. Oracle loads this post/wait algorithm into the AIX kernel to minimize overhead and to improve the performance. For Oracle Version 6, Oracle has implemented different kernel extensions (see the PW-README file in the Oracle install directory for more information).

    The rootpre.sh script copies Oracle kernel extension files to the /etc directory and modifies /etc/inittab file to automatically load the kernel extensions on system restart.

- Install Oracle software:

  After the Installer has been installed, the database administrator can use the Installer to copy Oracle products from the distribution media to the ORACLE_HOME directory.

- Relink:

  During Oracle installation or software maintenance, there is a procedure to regenerate Oracle programs by linking Oracle components with libraries. You have to relink Oracle programs if, for example, you:

  – Install SQL*Net

  – Change the dba group or use a group name other than the default dba

– Install software patches (fixes)

The relinking process will call library manipulation utilities to link Oracle programs with other Oracle products or libraries in order to generate the new executable program.

- Create database objects:

In the RDBMS server installation, the Installer can create a database with default parameters for you. Or the database administrator can create the database manually by using SQL commands.

In Oracle Development Tools installation, most of the tools will create special tables in the database to keep information about user applications. For example, SQL*Forms has tables in the database to record forms created by a user and the components in those forms.

- Run the post-installation script:

After the database is created, the Oracle Installer will create a shell script called root.sh. This script must be run by root in order to update Oracle configuration files in /etc and /usr.

5.3 Planning the Migration

This section discusses several considerations prior to the migration. First of all, before starting your migration, you must make sure that all Oracle products you use in your environment can run properly on AIX V4.1. Also, all your applications must be able to run on AIX V4.1. You might have to upgrade Oracle products and make some modifications to your applications.

We then discuss how long it takes to perform an entire system migration and who must participate in the migration process. We also discuss the migration methods and the main steps to be taken to perform the migration.

5.3.1 AIX and Oracle Levels

The first planning step is to check prerequisite versions of Oracle products to ensure that Oracle products will work properly on AIX V4.1, and make sure that they are officially supported by Oracle. Oracle RDBMS versions that are supported on AIX are shown in Table 19 on page 232.

```
┌─ Attention! ──────────────────────────────────────────────────────────┐
│                                                                        │
│  Before starting any migration, we recommend that you contact Oracle   │
│  to check the official support of any Oracle product you need.         │
│                                                                        │
└────────────────────────────────────────────────────────────────────────┘
```

From the table, you can see that AIX V4.1 is officially supported by Oracle V7.1.4 or later. Oracle V7.1.6 is the first release that supports AIX V4.1.3.

Note: Check with Oracle for a possible support of Oracle V7.1.4 in AIX V4.1.4.

In our testing environment, we used AIX V4.1.3 with Oracle V7.1.6 on both uniprocessor (model 580) and SMP (model J30) machines. So, in this chapter, most of the examples will refer to AIX V4.1.3 and Oracle V7.1.6.

Even though AIX V4.1 maintains binary compatibility with AIX V3.2, and even though most applications can be moved and run on AIX V4.1 without recompilation, we recommend that you migrate to Oracle V7.1.6 or later.

Table 19 (Page 1 of 2). Oracle RDBMS Versions Supported by AIX					
Oracle Version	**CDE Version**	**AIX Version**	**Comments**	**Status**	**HACMP Version**
6.0.31.0.1		3.1 Build 3005		Not shipping	
6.0.31.0.10		3.1	Patch #7 release from QA	Not shipping	
6.0.31.0.8		3.1	Patch #6	Not shipping	
6.0.31.0.8		3.2.0 - 3.2.2	Supported, except for CASE designer and Oracle Graphics	Not shipping	
6.0.34.2.1		3.2.0 - 3.2.2		Available for shipping	
6.0.34.3.1		3.2.0 - 3.2.2	Patch for 6.0.34.2.1	Not shipping	
6.0.36.3.2		3.2.0 - 3.2.3	Production	Not shipping	
6.0.36.7.1		3.2.0 - 3.2.3	Fix 1. CASE is not certified	Not shipping	
6.0.37.3.1		3.2.3 - 3.2.5		Not shipping; supported	
6.0.37.6.1		3.2.3 - 3.2.5		Shipping shortly	
6.0.37.8.1		3.2.5		Shipping	
6.2/7.0.x			Working on HA6000, UNIX Clusters	Not shipping	
7.0.11.2.2		3.2.3	Developper's release	Not shipping	
7.0.12.1.0		3.2.3	Production	Not shipping	
7.0.12.2.0		3.2.3	Patch 1	Not shipping	
7.0.13.1.0		3.2.3	8mm and 1/4 cartridge	Not shipping	
7.0.15.2	CDE P1	3.2.3	8mm and 1/4 cartridge	Not shipping	
7.0.15.4		3.2.3 - 3.2.5P	8mm and 1/4 cartridge	Not shipping	
7.0.16.4.0	CDE P2	3.2.3, 3.2.5	8mm and 1/4 cartridge	Not shipping	1.2, 2.1

Table 19 (Page 2 of 2). Oracle RDBMS Versions Supported by AIX					
Oracle Version	**CDE Version**	**AIX Version**	**Comments**	**Status**	**HACMP Version**
7.1.3	CDE P2	3.2.5	8mm and CD-ROM	Not shipping	2.1
7.1.4	CDE P3	3.2.5, 4.1.2	8mm and CD-ROM; OSI Protocol Adaptor is beta; 4.1 UP only; SMP is beta	Shipping	2.1
7.1.6	CDE P3	3.2.5, 4.1.2, 4.1.3, 4.1.4	8mm and CD-ROM	Shipping	2.1, 3.1*
7.1.6.2	CDE P3, DEV 2000	3.2.5, 4.1.2, 4.1.3, 4.1.4			2.1, 3.1*
7.2.2	CDE P3	3.2.5, 4.1.2, 4.1.3, 4.1.4			
7.2.2.4		3.2.5, 4.1.2, 4.1.3, 4.1.4		Patch Release	
7.2.3	DEV 2000	3.2.5, 4.1.2, 4.1.3, 4.1.4			

Note: ⋆ Oracle V7.1.6 and V7.1.6.2 support HACMP V3.1, with some restrictions. Please contact Oracle support for more details.

If you have Oracle V7.1.4 or later installed on your machine, you can migrate AIX to V4.1, and keep the existing Oracle software and database. If you have an earlier version of Oracle, you have to migrate both AIX and Oracle. Since Oracle V7.1.6 can run on both AIX V3.2.5 and AIX V4.1.3, we recommend that you migrate Oracle to V7.1.6 first, then upgrade AIX to V4.1.3. It is safer to migrate Oracle and AIX separately because it is easier to determine the origin of a problem. Also, it is easier to go back to the previous system level if necessary.

If you have AIX V3.2.5 with Oracle V6.0 or V7.0 installed, you can do the following:

- Upgrade Oracle to Version 7.1.6.

  Once you have Oracle V7.1.6 running, you can test your application, and make sure that everything works correctly. At this time, you can keep two versions of Oracle on the same machine. You can go back, and use the older version of Oracle by restoring the database and changing the configuration. It is not necessary to reinstall AIX. After you have finished all your tests, you can go to the next step:

- Upgrade AIX to Version 4.1.3.

  Once AIX is upgraded to AIX V4.1.3, you can test your application. At this step, the only way to go back to AIX V3.2 is to restore your system backup.

5.3.2 Checking Oracle Products Level

Oracle has several product offerings on AIX ranging from Oracle RDBMS server and application-development tools to CASE designer. The Oracle RDBMS is usually bundled with many Oracle products.

Table 20 shows the content of the Oracle RDBMS 7.1.4 bundle which is supported in both AIX V3.2.5 and AIX V4.1.2.

Table 20 (Page 1 of 2). Oracle RDBMS 7.1.4 Bundle	
Product	**Version**
Oracle DBMS	7.1.4.1.0
PL/SQL	2.1.4.0.0
Distributed Option	7.1.4.1.0
Parallel Query Option	7.1.4.1.0
Pro*C - XL C	2.0.4.0.0
Pro*C - XL C	1.6.5.0.0
SQL*Module C - XL C	1.0.4.0.0
Oracle Server Manager	2.0.3.0.0
SQL*Plus	3.1.3.5.1
SQL*Forms up to V3.0	3.0.16.12.9
SQL*Menu	5.0.11.13.5
Oracle Reports	1.1.14.14.1
Easy*SQL	2.2.26.0.4
SQL*Calc	1.2.25.0.1
SQL*Report	1.1.10.3.1
SQL*QMX	1.0.26.0.1
Oracle Data Query	3.1.0.7.1
Multi-Media APIs (MM)	1.0.7.1.2
Oracle*Help	1.0.10.1.1
Oracle Network Manager	2.1.4.1.0
Oracle Network Manager	2.1.4.1.0
SQL*Net Async	1.2.13.2.1
SQL*Net TCP/IP	2.1.4.1.0
SQL*Net TCP/IP	1.2.7.8.1
Multi-Protocol Interchange	2.1.4.1.0
SQL*Net	2.1.4.1.0
CASE*Designer	1.1.21.8.1
CASE*Dictionary for RDBMS V5	5.0.22.9.0
Oracle Forms Generator	2.0.8.2.0

Table 20 (Page 2 of 2). Oracle RDBMS 7.1.4 Bundle	
Product	**Version**
Oracle Reports Generator	1.0.9.2.0
Oracle*Mail Server	1.1.3.9.2
Oracle*Installer	3.0.10.1.3
Oracle*XA	1.1.1.0.0

Table 21 gives the content of the Oracle RDBMS 7.1.6.2 bundle which is supported in AIX V3.2.5, AIX V4.1.2, AIX V4.1.3, and AIX V4.1.4.

Table 21 (Page 1 of 2). Oracle RDBMS 7.1.6.2 Bundle	
Product	**Version**
Oracle DBMS	7.1.6.2.0
Oracle Parallel Server Option	7.1.6.2.0
Distributed Option	7.1.6.2.0
Parallel Query Option	7.1.6.2.0
Oracle Symetric Replication Facility	7.1.6.2.0
Pro*Ada	1.6.7.0.0
Pro*FORTRAN	1.6.7.0.0
Pro*COBOL	1.6.7.0.0
Pro*C	2.0.6.0.0
Pro*C	1.6.7.0.0
SQL*Module for Ada	1.0.6.0.0
SQL*Module for C	1.0.6.0.0
Oracle Server Manager	2.1.3.0.0
SQL*Plus	3.1.3.7.1
SQL*Forms/Menu	3.0.16.12.9
SQL*ReportsWriter	1.1.14.15.1
Oracle Data Query	V3.1.0.7.1
Oracle Protocol Adaptor for OSI - OSSC	2.1.6.1.0
Oracle Async Protocol Adapter	1.2.13.2.1
Oracle DECnet Protocol Adapter	2.1.6.1.0
Oracle DECnet Protocol Adapter	1.2.18.2.1
Oracle SPX/IPX Protocol Adapter	2.1.6.1.0
Oracle SPX/IPX Protocol Adapter	1.2.7.9.1
Oracle TCP/IP Protocol Adapter	2.1.6.1.0
Oracle TCP/IP Protocol Adapter	1.2.7.8.1
Oracle APPC Protocol Adapter	2.1.6.1.0
Oracle APPC Protocol Adapter	1.1.0.1.6
Multi-Protocol Interchange	2.1.1.0.0
SQL*Net	2.1.6.1.0

Table 21 (Page 2 of 2). Oracle RDBMS 7.1.6.2 Bundle	
Product	**Version**
SQL*Net	1.2
CASE Designer	1.1.21.8.1
CASE Dictionary	5.0.22.9.0
CASE Generator for SQL*Forms & SQL*Menu	2.0.8.2.0
CASE Generator for SQL*ReportWriter & SQL*Plus	1.0.9.2.0
Oracle*Mail Server	1.1.3.9.2
Oracle*Mail Client	1.1.3.9.2
PL/SQL	2.1.6.2.0
Codegen	7.1.6.2.0
Slax	7.1.6.2.0
Sqllib	1.6.7.0.0
PRO*CORE	7.1.6.2.0
NLSRTL	3.1.4.3.0
NLSRTL	2.3.6.0.0
Oracore	3.4.1.0.0
Oracore	2.3.7.1.0
Easy*SQL	2.2.26.0.4
SQL*Calc	1.2.25.0.1
Oracle Toolkit II - MOTIF	2.1.4.5.0
Oracle Toolkit II	2.0.10.22.0
Oracle*Help	2.1.1.0.0
Oracle*Help	1.0.10.1.1
Multi-Media APIs (MM) - MOTIF	2.0.5.4.0
Multi-Media APIs (MM)	1.0.7.1.2
Oracle Names	1.0.5.0.0
Declarative Reusable UI Drive (DRUID)	1.1.5.0.0
OACORE	2.1.3.0.0
ORATST	2.1.5.0.0
Makeship	3.0.4.0
CRT	1.0.11.0.3
ORAKIT (Object*SQL)	1.0.20.3.1
Oracle Terminal	1.0.10.0.4
Oracle*XA	1.1.1.0.0
Oracle*Installer	3.0.10.1.4
SQL*Report	1.1.10.3.1
Oracle Network Manager	2.1.6.0.0

If you need the list of products included in a specific RDBMS bundle, please contact your Oracle support.

> ┌─ **Attention!** ───
> │
> │ Oracle products that are not part of the RDBMS bundle might not be supported
> │ in AIX V4.1. Please verify with your Oracle support that all the Oracle products
> │ you are using are supported in the version of AIX V4.1 you are migrating to.
> │
> │ Also, if you are using Oracle applications, check if your level is currently
> │ supported in AIX V4.1.
> │
> └──

5.3.3 Oracle Upgrade Paths

Table 22 shows how to migrate from different versions of Oracle to Version 7.1.6.

If you have already installed Oracle V7 and plan to install a newer release, Oracle has provided an upgrade utility to upgrade your database. The upgrade utility will make some modifications to the database structure to support the new release while maintaining existing data and configuration. This procedure is called "upgrading". The upgrading process does not require a database backup and restore. However, once the database has been upgraded to a newer version, the only way to go back to the older version is to restore the database. So, you should back up the database before making any change.

For Oracle V6, the procedure to move data from V6 database to V7 is called "migration". You have to perform a backup, export the whole database to a backup media, install Oracle V7 distribution, recreate the database, and then import data back to the system.

Another alternative to migrate from V6 to V7 is to use the Migration Utility. Oracle V7 provides a utility to quickly migrate the entire database from V6 to V7. This utility will convert some files and database structures to the Version 7 format. The Migration Utility takes less time than an Export/Import and consists of fewer steps.

For Oracle V5 or earlier, you must migrate to Oracle V6 before you can migrate to V7. See the *Oracle V6 Installation and User's guide* for more information.

Table 22. Oracle Migration Paths to Oracle V7.1.6		
Version	**Release**	**Methods**
V5	Any release	Migrate to V6 then migrate to V7
V6	6.0 up to 6.0.26	Perform Export/Import migration
	6.0.27 or later	Use the Migration Utility
V7	7.0 up to 7.0.10	Perform Export/Import migration
	7.0.11 or 7.1.3	Upgrade with Installer
V7.1	7.1.4 or later	Supported by AIX V3.2.5 and AIX V4.1.2

5.3.4 AIX V4.1 New Packaging

In AIX V3.2, the XL C compiler is included in the AIX standard package. The new packaging of AIX V4.1 does not include the C compiler, but it is shipped in another license program product, C Set++ V3.1. You will have to order the C Set++ if you use the following products or features:

- Pro*C
- SQL*Module for C

Or if a user has created C exit routines in SQL*Forms

During Oracle installation, there is a process to relink Oracle programs. In older versions of Oracle, the relinking process requires the C compiler to perform this task. However, in Oracle V7.1.6, it uses library manipulation utilities, such as make, ar, ld and so on, that come with AIX V4.1. So, you do not need the C compiler to relink Oracle V7.1.6.

5.3.5 Third Party Software

If you have a third party application installed on the system, you should contact the corresponding vendor to verify if this application is supported on AIX V4.1.

5.3.6 Estimating the Migration Time

During the migration, all the applications and databases must be stopped. Users will not be able to access the information inside the database for a long time. Here are some major steps that you may use to estimate the time for the whole migration process.

- Upgrading to AIX V3.2.5:

  If you have an older version of AIX V3.2 installed, such as AIX V3.2.2 or AIX V3.2.3, you have to upgrade the machine to AIX V3.2.5. This procedure may take several hours, depending on the number of Licensed Program Products (LPPs) installed and on the installation device. An estimated time for AIX V3.2.5 Preventive Maintenance Package (PMP) installation is around 4-5 hours.

- Backing up the AIX V3.2.5 system:

  Backup time depends on the amount of data in the AIX filesystems, on what kind of backup device you use and how the device parameters are set, such as block size, compress algorithm and so on. See the *Storage Systems Marketing Guide, RS/6000 and Selected UNIX Systems* for general guidelines about the speed of each backup device.

  Note: For IBM employees, this guide is available in MKTTOOLS. The name of the package is RS6KSTOR.

- Backing up the databases:

  Back up time depends on the size of the database and the type of backup device used for the backup. You have to make an AIX system backup and a database backup separately if your database is on raw devices since the AIX system backup does not backup data in raw devices. Also, this will allow you to restore only the database without touching any other AIX files if there is a problem during the Oracle migration.

  Usually, the database is the most important part in the system. You may want to have several copies of the database backup or use several backup methods,

such as backing up data files and exporting a database. So, backing up the database may take a lot of time.

- Installing a new release of Oracle:

  This step moves Oracle code from the installation media to disk. Since Oracle has several product offerings, the installation time may vary, depending on the number of products to install.

- Upgrading Oracle:

  If you have an old version of Oracle, you have to upgrade Oracle. The time for upgrading database objects depends on the upgrade method and the size of your database. Upgrading with Installer is the easiest method and takes less time, while upgrading with Import/Export utilities is the most complicated method and takes the longest time.

  You have to repeat the upgrade procedure for each database in your system; so you have to reserve time to upgrade each database.

- Migrating AIX:

  The AIX migration time will vary by number of LPPs installed. It also depends on the installation device. For more information, refer to Chapter 1, "Migrating AIX from V3.2.5 to V4.1" on page 3 in this redbook.

- Testing the new environment:

  After the migration, application developers and end users have to test their applications in the new system. This step may take a lot of time. If you have a good plan on how to test the applications in advance, it will reduce the time for testing.

- Recovery:

  You have to reserve time for recovering the system if something goes wrong during the migration.

The purpose of the information below is just to give you some ideas about the migration time. Table 23 on page 240 shows some steps in the migration and the time to perform each step on the testing machine. The table does NOT show some major steps that may take a lot of time, such as planning for the migration, upgrading the application, testing the application, recovery time, and so on.

This is the configuration of our testing machine:

- RISC System/6000 model 580 with:
  - 5 GB tape drive (internal)
  - CD-ROM drive (internal)
- AIX V3.2.5 with:
  - Base Application Development Tools (bosadt)
  - Extended Commands (bosext1)
  - Network subsystem: TCP/IP, NFS (bosnet)
  - XL C compiler (xlccmp)
  - AIXwindows
- Oracle V6 and V7 RDBMS with:

- PL/SQL

- SQL*Plus

- SQL*Forms

- SQL*Menus

- SQL*Report

- SQL*Net

- Four small databases (40-80 MB) in raw devices and UNIX files.

Note: Your results may vary. The migration time will depend on your system configuration.

Table 23. Time for Each Migration Step	
Description	**Time**
AIX system backup (around 800 MB of data in rootvg file systems using a 5 GB tape drive)	45 minutes
Database backup	Depends on the database size
Install Oracle V7.1.6 from CD-ROM	2 hours
Upgrade a V7 database to V7.1.6 using the Installer (for each database)	1 hour
Migrate a V6 database to V7.1.6 using the Migration Utility (for each database)	2-3 hours
Install AIX V4.1 from CD-ROM (using the Migration Install)	1:45 hours

5.3.7 Who Should be Involved in the Migration?

On the system, you may have Oracle and several applications installed. Each component requires skilled personnel to manage and solve the problems. Here are general guidelines about personnel who should get involved in the migration process and their roles.

1. AIX system administrator:

   The system administrator's responsibilities are:

   - Develop an overall migration plan and perform the migration.

   - Identify AIX configurations, such as network configuration, filesystems, volume groups, devices, and so on.

   - Backup the AIX system.

   - Restore the AIX system if a recovery is necessary.

2. Database administrator:

   The database administrator or dba should have a basic skill in managing the database and backing up/restoring data. He/she should have a good understanding of the existing database configuration in the machine.

   The database administrator's responsibilities are:

   - Develop Oracle upgrade procedure.

   - Identify Oracle database configuration.

- Backup the database.

- Perform the Oracle upgrade.

- Restore the database if a recovery is necessary.

3. Application developer:

   You also need a developer or a user who has a very good understanding of the application. The person should know the main components and functions of the application and be able to use the application effectively in order to test it.

   The application developer's responsibilities are:

   - Develop a testing plan. Test the main features of the application, and make sure that the application works the same way after the migration.

   - Identify changes in the applications. You may have to upgrade Oracle development tools that will affect some applications. The developer may have to make modifications in the application source code in order to support the new version of Oracle.

5.3.8 Developing a Testing Plan

To ensure that the database and the application work in the same way after the migration, you should develop a testing plan.

- List main procedures that the user will do in normal operation.

- Test them on an AIX V3.2 system, and record the results.

- After the migration, run these tests again, and compare the results.

5.3.9 Practicing on a Testing Machine

In a mission-critical environment, you should try to run a first migration on a testing or development machine. After you have tested that all applications work correctly in AIX V4.1, then migrate your production machine, and apply the knowledge that you have found on the testing machine.

If a customer has only one machine, you may try to create a system backup to clone the machine, and do all the testing. If there is only one machine available, Oracle has a feature to create multiple databases (or instances) on the same machine. You can create a new database by using a small set of data; migrate the database, then do the testing on this database before making any changes to the production database.

5.3.10 Planning for Recovery

For both AIX and Oracle migration, we try to preserve user information as much as possible and, at the same time, try to eliminate steps to backup/restore data. However, things may go wrong, and you must be able to recover the system. So, you must have the necessary backup data and a good procedure to recover the system.

In the previous section, we discussed that in some environments, you have to migrate both AIX and Oracle. You have to prepare for recovery from either an AIX or an Oracle migration failure.

5.3.10.1 Recovering After an AIX Migration Failure

Once you have installed AIX V4.1, the only way to go back to AIX V3.2 is to reinstall the system. The best way to guard against an AIX migration failure is to use the system backup utility in SMIT or use `mksysb` from the command line.

The system backup utility will backup all files in the AIX journaled file systems inside rootvg. It also creates a boot image on the backup media so this media can be used to boot the system.

```
  ┌─ Attention! ──────────────────────────────────────────────────────┐
  │                                                                    │
  │  AIX V3.2.5 system backup will NOT backup a raw device and also    │
  │  will NOT backup files in non-rootvg volume groups.                │
  │                                                                    │
  └────────────────────────────────────────────────────────────────────┘
```

When you restore the system, the restoration will overwrite everything in the old rootvg and recreate a new one. So, if you have data in raw devices, you have to back them up separately.

Information stored in non-rootvg volume groups will be preserved during the migration. However, to protect information from a disk failure or a human error (for example: the operator accidentally keys the wrong command and destroys the data), you can back up data in non-rootvg by using AIX utilities such as tar, cpio or backup.

5.3.10.2 Recovering After an Oracle Migration Failure

Oracle files can be separated into three main parts: Oracle code, database and application. Refer to section 5.2.1, "Oracle Files Organization" on page 227 for more information about these components.

In Oracle migration, you can install a new version of Oracle in a separate directory so you can keep the old and the new version of Oracle at the same time. You can select which version of Oracle to use by changing configuration files. By the way, if you install Oracle in rootvg, Oracle code will already be included in the system backup. You can extract only the Oracle code from a system backup tape. In the worse case, if you lose the Oracle code, you can just reinstall it from the Oracle distribution media.

The database will be updated (or recreated) during the Oracle migration. If you want to go back to an older version, you have to restore the database. Oracle databases can be backed up with AIX utilities or with Oracle Export/Import utilities. Section 5.6.1.3, "Backing up an Old Database" on page 253 describes how to use these utilities. You may want to use both methods to complement each other. Table 24 compares the differences between these two backup methods.

Table 24 (Page 1 of 2). Backup Methods Comparison		
Backup Method	**Advantages**	**Limitations**
Backup data files	• Easy to restore. • Maintains database configuration.	Must be restored to: • The same database. • The same version of Oracle. • The same datafiles.

Table 24 (Page 2 of 2). Backup Methods Comparison		
Backup Method	**Advantages**	**Limitations**
Export/Import	Flexibility to restore: • Only some users or some tables. • To a different database. • To a later Version of Oracle.	During import: • It requires the same tablespaces. • Tablespaces may overflow.

For the applications, some of them can run on a newer version of Oracle without any changes. Some of them have to be updated and recompiled to support the new release. If you have to migrate the application or install the new version of the application, you should backup those files. See the *Oracle Migration Guide* for more details.

5.3.11 Client/Server Environment

In a complex environment with multiple machines, you can develop a migration plan to move small group of machines at a time in order to reduce the risk.

In a client/server environment, the server machine contains the database server, and the client machines contain the application. Client and server machines use SQL*Net to communicate with each other.

Oracle tries to maintain compatibility between Oracle versions. The new version of Oracle RDBMS usually has the capability to support existing applications. On the other hand, the new release of an application may contain new commands or new syntaxes that are only supported by the new RDBMS server. These new commands and syntaxes cannot be used with the old release of RDBMS server.

From the above limitation, the server machine should be upgraded first. You can keep client machines running an older version, and test the new version on the server machine. After you have done all the testing on the server, you can migrate the client machines to AIX V4.1.

However, you should contact Oracle to verify that Oracle products on your client machine will work with the new Oracle V7.1.6 when you upgrade the server.

Oracle has two versions of SQL*Net, Version 1 and Version 2. SQL*Net V2 has several new features and should be used for new installations. SQL*Net V1 for TCP/IP is still supported on AIX V4.1; so a customer who has SQL*Net V1 for TCP/IP can continue to use this software on AIX V4. However, if you plan to use the new features of SQL*Net V2, like the multiprotocol support, you should plan to upgrade SQL*Net to V2 in the future.

5.4 Methodology

This section describes the main steps in the migration process.

5.4.1 Overview of the Overall Migration

The migration procedure depends on the existing configuration and version of software installed on your machine. Figure 49 on page 245 shows an overview of the migration process. We chose to migrate the software step by step, and test the environment after each migration instead of upgrading everything at once and running all the tests at the end. We did so because it is safer to change only one component at a time and easier to determine the origin of a problem. Also, it is easier to go back if there is something wrong.

We suggest the following methodology:

- Upgrade AIX to V3.2.5:

  If your machine is already running AIX V3.2.5, you can skip this step. The rest of this chapter will assume that your machine has AIX V3.2.5 installed. The reason to upgrade AIX to V3.2.5 first comes from the fact that Oracle V7.1.6 requires that AIX must be at least V3.2.5. For AIX V3.2.4 or earlier, this step may take a lot of time, but it will not cause major changes to AIX. Most applications should be able to run without any problem. However, we recommend that you check with both Oracle and third-party software vendors to ensure that your application will run on AIX V3.2.5.

  See the *AIX V3.2.5 Release Notes* for information on how to upgrade your system to AIX V3.2.5.

- Migrate Oracle to V7.1.6:

  Second step, if you have an older version of Oracle, you must upgrade Oracle to V7.1.6. You can keep both versions on the same machine. After upgrading to a new release, you can test the applications, and remain in this environment until you want to migrate to AIX V4.1.

- Migrate AIX to V4.1:

  After you have finished all the tests, you are now ready to upgrade AIX to V4.1.
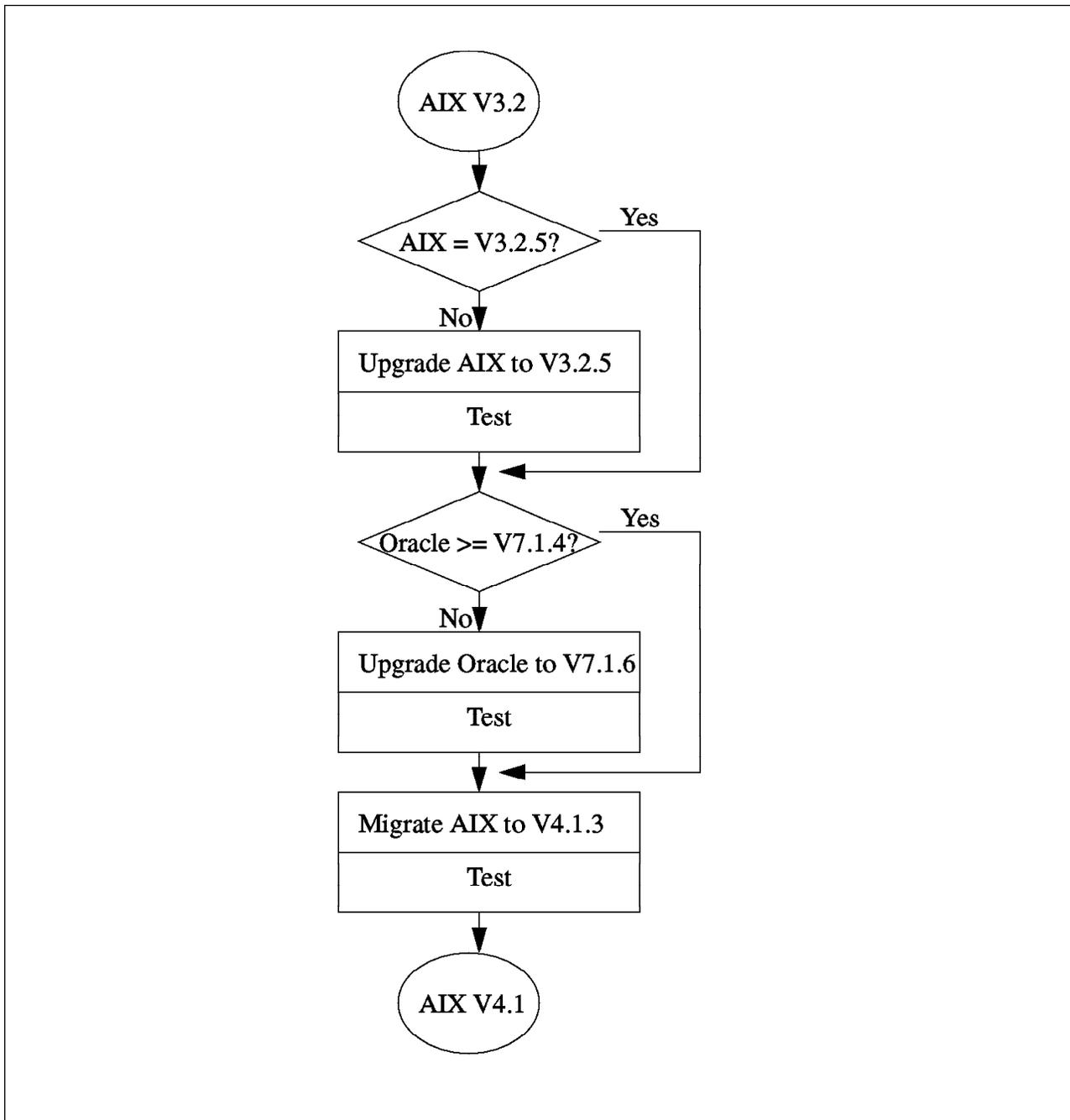
Figure 49. AIX and Oracle Migration Flowchart

5.4.2 Oracle Migration Overview

Oracle provides upgrade paths from an earlier version to Oracle V7.1.6. The upgrading methods depend on the version of Oracle installed on the machine. Before you make any change to the database, the whole database must be backed up to protect information from a migration failure.

There are three main methods used to upgrade Oracle: using the Installer, using the Migration Utility and using Export/Import utilities.

1. The Installer is used to upgrade the Oracle V7 database object to support the new release.

2. The Migration Utility is used for Oracle Version 6 to Version 7 migration.

   Both of these methods change the database objects directly and do not require export/import data.

3. Using Export/Import utilities. All data in the old database must be exported to a backup media. Then a database administrator must create the new database and import the data back into the database. This method takes more time and is comprised of more steps, but this method can be applied to all versions of Oracle. The Import utility also provides more flexibility to selectively restore some parts of the database.

After upgrading the Oracle database, if there is a problem with the new version and you need to go back to the old version, you must restore the entire database.

Figure 50 shows different methods of migrating Oracle.
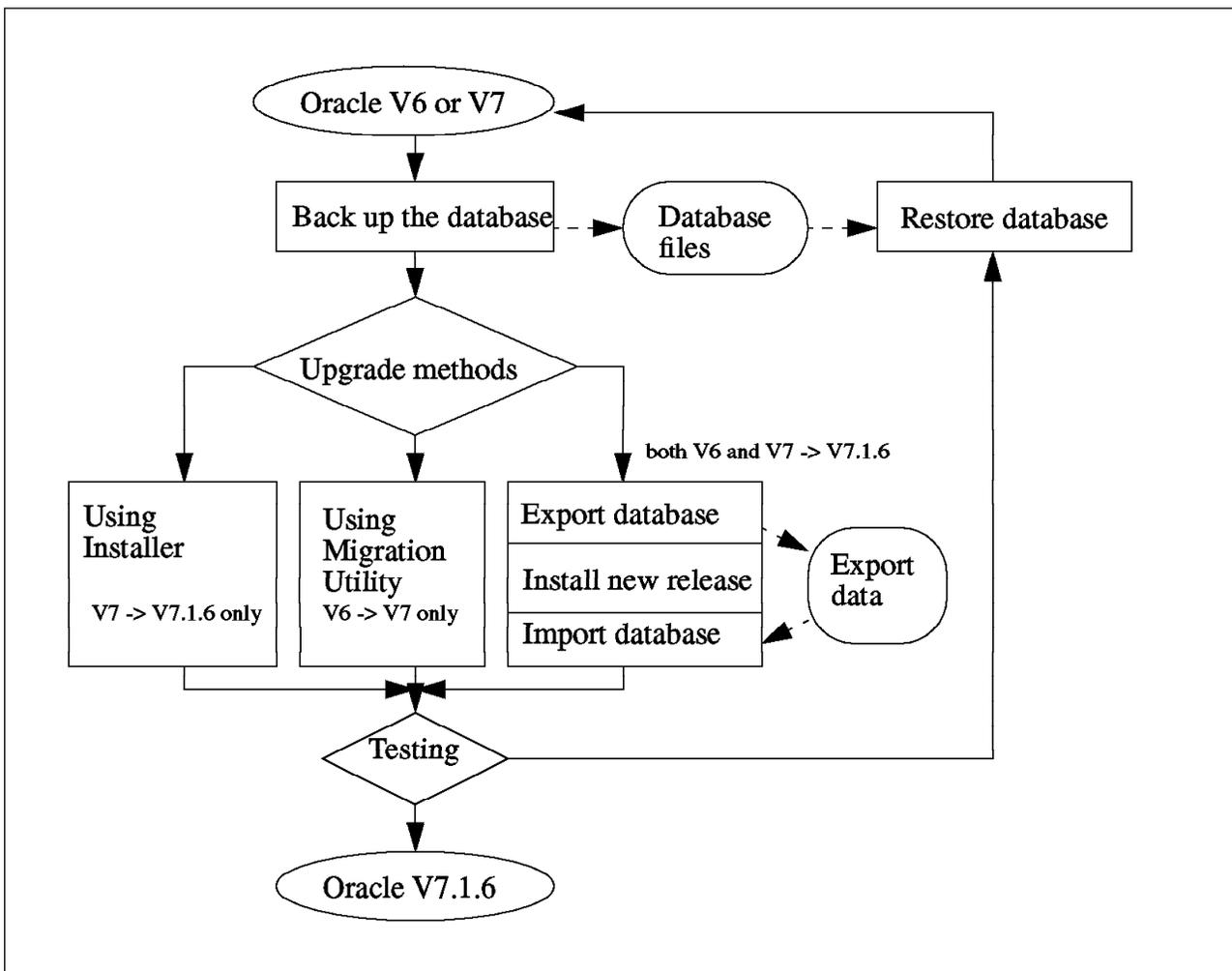
Figure 50. Oracle Migration Flowchart

5.4.3 AIX Migration Overview

AIX migration with Oracle V7.1.6 has the following steps:

- Documenting the system:

  This step collects information about existing databases and related files in order to prepare for backing up.

- Backing up the system:

  After you have all the information about the databases, you can start backing up databases files, and then back up all other AIX files by using the system backup utility.

- Migrating AIX:

  This step installs AIX V4.1 into the system. The AIX migration replaces the AIX kernel and files with the new version of AIX and makes some modifications to the existing configuration files.

- Post-migration steps:

  After rebooting the machine in AIX V4.1, you have to make some changes to AIX configuration and call Oracle scripts before the database can be restarted again.

- Testing:

  Now the system is running in the new environment. Application developers and users have to test all the applications to make sure that everything works properly. After the testing, you have to make the decision to keep this environment or go back to the earlier version.

- Recovery:

  If there is a problem with the new system, to restore the system back to the previous working environment, you have to:

  - Restore AIX: This step will restore all AIX files back to the system. However, the restoration will not preserve raw devices.

  - Restore raw-devices: If a customer has a database in raw device files, you have to recreate and restore them separately.

  - Post-restore step: This step will check information in non-rootvg volume groups and make a necessary change to complete the recovery.

Figure 51 on page 248 is a flowchart showing how to upgrade AIX using the Migration Install method, how to migrate from AIX V3.2 to AIX V4.1.3 with Oracle V7.1.6 and how to recover back to AIX V3.2 if there is a problem with V4.1. It also shows how the backup data will be used.
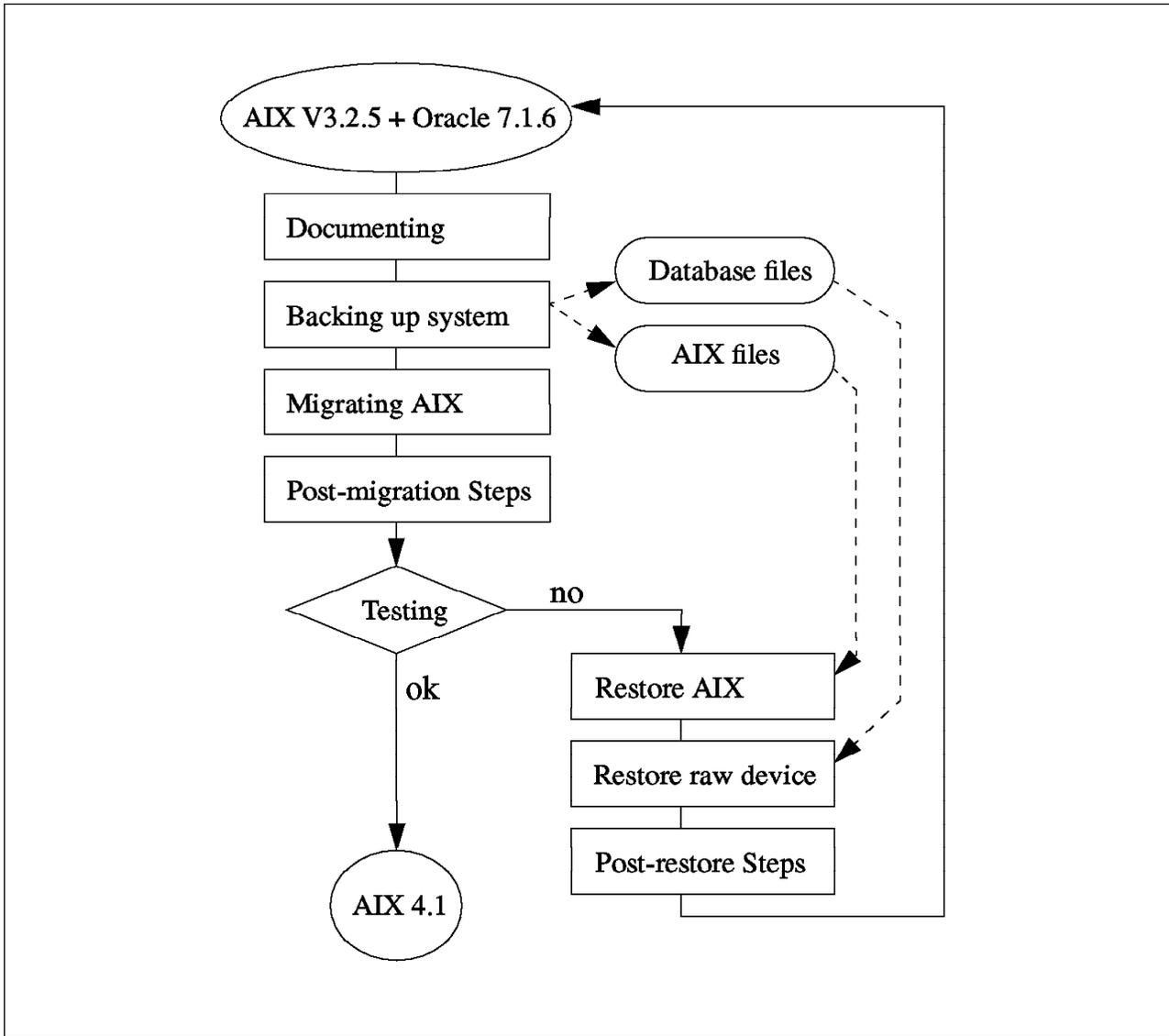
Figure 51. Migrating AIX with Oracle V7.1.6

5.5 Migration Checklist

Here are some general guidelines that you should complete before the migration.

- Identify all Oracle products, custom and third-party applications that are installed on your machine.

- Verify that all applications using Oracle have been ported and tested previously for Oracle V7.1.6 and AIX V4.1.3.

- Develop a migration plan. This plan should include:

  - A testing plan: how to test your application in the new environment and what the results of the test should look like.

  - AIX and Oracle upgrade plan: how to install each software and how to upgrade the existing configuration.

  - Recovery plan: what to do if something goes wrong.

- Prepare all necessary resources:
  - AIX V4.1.3 installation media
  - Oracle V7.1.6 installation media
  - AIX and Oracle manuals
  - Your technical staff, such as the system administrator, the database administrator and the application developer(s)
  - Installation and back up devices (plus associated blank media) , such as 8mm tape, CD-ROM, and so on

5.6 Upgrading Oracle V6 and V7 to V7.1.6

This section discusses the various methods used to upgrade Oracle to V7.1.6 and gives detailed procedures for each upgrade method.

In the upgrade process, the new release of the Oracle product is installed into the system. You can decide to install the new release in either the old ORACLE_HOME directory or in a new directory. Oracle recommends creating a new directory and install the new release separately from the old release. So, you can keep both releases at the same machine and have two databases with different releases running at the same time.

After the new release of Oracle is installed, you have to upgrade the database objects. This step makes a change to the configuration files and data files of the database in order to support the new release. This step must be performed to each database in the system, one by one. You can start with the testing or development database, and run the test to ensure that the migration will not cause any problem. Then you can start upgrading the production database.

There are three migration methods that are covered in this document: using the Installer, using the Migration Utility and using the Export/Import utilities. They are described in the three following sections, and the last section discusses the recovery process from a migration failure.

This section describes the main procedure in the migration and may not cover all the details. Please consult your local Oracle support, and refer to *Oracle7 Server Migration Guide* and *Oracle7 for AIX-Based Systems Installations and Configuration Guide* for more details.

5.6.1 Upgrading Oracle V7 to V7.1.6 on AIX V3.2.5

If Oracle V7 has been installed on AIX V3.2.5 and you want to migrate to Oracle V7.1.6, you can use one of the following methods.

1. Oracle Installer:

   This method will apply to an Oracle V7.0.11 or later. The Installer has an option to upgrade the database objects to a newer version. Detailed instructions will be discussed later in this section.

2. Control file rebuild:

   Oracle provides a utility called upgrade709 to update the control file of the V7.0.9 database to the current release without performing an Export/Import.

However, this book will not cover this method; please consult Oracle, or see *Oracle7 Server Migration Guide* for more details.

3. Export/Import:

   This method can apply to any release of the Oracle server. The Export utility will extract data from the database and put it into a single file. You have to reinstall the new version of the Oracle server, recreate the database and then use the Import utility to restore the data back to the database. This method is discussed in section 5.6.3, "Migrating Oracle V6 or V7 to V7.1.6 Using Export/Import Utilities" on page 262.

If you have *Oracle V7.0.11 or later* installed, follow the following steps to upgrade it to Version 7.1.6.

1. Document the system.

2. Shutdown the old database.

3. Backup the old database.

4. Modify a raw device (if any).

5. Change the environment settings.

6. Install the Oracle V7.1.6.

7. Copy and modify configuration files.

8. Upgrade the database objects.

9. Update the configuration in /etc.

10. Test.

5.6.1.1 Documenting the System

You must have complete information about the current configuration of your system in order to back up databases properly. These are the things you need to know:

- Number of databases in the system and the name of each database

- Configuration files for every database

- Location, size and type (raw devices or UNIX files) of every file related to the databases

The following instructions describe how to collect this information:

Identify all Databases in the System: A database administrator can configure several Oracle instances on one machine. One instance contains only one database, which is comprised of a configuration file and several data files. Usually, the instance name and the database name are the same. In Oracle, the instance name is called System Identifier and is usually referred to as the SID.

Look at the file /etc/oratab which contains a list of Oracle databases configured on the machine.

For example:

```
PROD:/u01/home/dba/oracle/product/7.0.15:Y
DEV1:/u01/home/dba/oracle/product/7.0.15:Y
TEST:/u01/home/dba/oracle/product/7.1.3:N
```

The first column is a database name or SID (four characters). The second column is the ORACLE_HOME directory where Oracle products have been installed. And the last column is a flag indicating whether or not to start a database with the database administrator call, dbstart. In this example, there are three databases: PROD, DEV1 and TEST.

In some cases, a database administrator may create a new database but incorrectly update the /etc/oratab file. Please consult your database administrator to verify that you have listed all the databases in the system.

Identifying all Files Related to a Database: These files are configuration files, control files, data files, and log files.

- Configuration files:

  There is a master configuration file, initSID.ora, for each database. It is located in the directory $ORACLE_HOME/dbs by default; but if the database administrator follows Oracle Optimal Flexible Architecture (OFA), it will be located in the $ORACLE_BASE/admin/SID/pfile directory. Oracle will read this file during the database startup process and initialize shared memory and server processes according to the parameters specified in this file. The initSID.ora file may call another file, usually named configSID.ora. Look for the parameter ifile in initSID.ora.

  For example:

  ```
  #
  # $Header: initx.orc 7001300.6 95/02/11 14:16:11 wyim Osd<unix>
  #
  # include database configuration parameters
  #
  ifile = /u01/home/dba/oracle/product/7.1.6/dbs/configPROD.ora
  ```

  You must have both the initSID.ora and configSID.ora files in order to start the database.

- Control files:

  Once you have found the initSID.ora and configSID.ora files, look for the stanza control_files in these two files. It points to control files of this database. You may have multiple copies of control files.

  For example:

  ```
  # cnfg.ora - instance configuration parameters

  control_files            = (/u01/ORACLE/PROD/control01.ctl,
                               /u02/ORACLE/PROD/control02.ctl,
                               /u03/ORACLE/PROD/control03.ctl)

  db_name                  = PROD
  ```

  This example has three copies of control files in three different paths: /u01, /u02 and /u03. Record all the path names of the control files.

- Data files and redo log files:

  To collect information about data files and log files, you have to access through the Oracle sqldba utility. The information is kept inside the Oracle system catalog.

Log in as the Oracle owner:

```
$ export ORACLE_SID=<database_name>
$ sqldba
SQLDBA> connect internal
SQLDBA> select name from v$dbfile;
SQLDBA> select member from v$logfile;
SQLDBA> exit
```

The sqldba utility will show a list of data files and log files on the screen. Record the path names of these files, and then use an AIX utility, such as the `ls -l` command, to check the size of each file, and identify whether the file is a normal AIX file or a raw device. The backup method for a raw device is different from the method for a normal file. Also, during the migration, you may have to modify data in the raw device.

- Script files:

  There are SQL script files, crdbSID.sql and crdb2SID.sql, that are used to create the database. They are usually located under the directory $ORACLE_HOME/dbs, but if you followed OFA rules, they will be located under the $ORACLE_BASE/admin/SID/create directory. Once the database is created, these script files are no longer used, unless you want to recreate the database again.

After you have recorded all the file names in the database, repeat these steps with the other databases until you have completed all the databases in your system.

5.6.1.2 Shutting Down Old Databases

After you have identified all configuration and data files, you need to stop the database to protect data from being updated while you perform a backup. All data files must be closed and be in a consistent state to ensure data integrity. Have all users disconnected from the database. Stop all the batch jobs that access the database.

Log in as the Oracle owner, and issue the following command to shut down all databases.

```
$ dbshut
```

OR shut down the databases one by one:

```
$ export ORACLE_SID=<database_name>
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> shutdown normal
SQLDBA> exit
```

If there are some users attached to a database, the Oracle server will wait until all users are disconnected. To force a user off the database, use the command `alter system kill session`. See sqldba help for more details.

5.6.1.3 Backing up an Old Database

To back up an old database, you can use either the AIX utilities or the Oracle Import/Export utilities to back up data files.

- **Using AIX backup utilities to back up data files**:

  This method uses standard AIX utilities to back up all data files from a hard disk to a backup media. All the files must be backed up from the same consistent state. It means that you have to shut down the database in order to stop all activities and prevent any update to those files. To restore data, you need the configuration files. You then need to use AIX utilities to transfer data files from the backup media back to the disk in the old directory, and then restart the database.

  - For database files that reside on a normal AIX filesystem. You can use `cpio`, `tar` or `backup` commands to back up data files that have been identified in the section "Identifying all Files Related to a Database" on page 251.

  - For raw device data files, use the `dd` command to copy data to tape or to another filesystem.

    For Oracle V7.0.16 or later, data will begin at the second block of the logical volume to avoid overwriting the AIX Logical Volume Control Block (LVCB). To copy data, use the following command.

    `# dd if=/dev/roradata of=<temp_file> bs=4096 skip=1`

    where /dev/roradata is the raw device name.

    For earlier versions up to V7.0.15, data will begin at the first block of the logical volume. To copy data, use the following command:

    `# dd if=/dev/roradata of=<temp_file> bs=4096`

    > **Note**
    >
    > All the files (control, data and redo log files) must be backed up from the same consistent state. If the database is reopened, every file must be backed up again.

- **Using Import/Export utilities to back up data**:

  The Export utility will extract data from each object in the database and convert it into one flat file. This method provides more flexibility to restore only some parts of the database. It is also used in the migration from Oracle V6 to V7.

  To restore data, you must recreate the database, which may have some different parameters from the original database. For example, you can create data files with different names and sizes. The Import utility will convert an export file and insert data back into the database. It requires that all tablespaces must exist in the database with sufficient space.

  To use the Export utility, restart the database in restrict mode. This will allow only the database administrator to connect to the database. Then, run the `exp` command.

  For example:

```
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> startup restrict
SQLDBA> exit
$ exp userid=system/<manager> file=<export_file> full=y
```

Note: Replace `<manager>` with your system account's password and
`<export_file>` with the backup device name or the temporary file.

5.6.1.4 Modifying Raw Devices

AIX uses the first block of a logical volume to keep a logical volume control block
(LVCB). Oracle V7.0.16 and later versions try to avoid overwriting this information
by skipping the first block and writing data on the second block of the raw device.
If you have Oracle V7.0.16 or later installed, skip to the next step.

If you have Oracle Version 7.0.15 or earlier and use a raw device, do the following
steps:

- Back up the old raw device:

  Issue the following command to back up the raw device to a temporary file or
  to a backup media.

  ```
  $ dd if=/dev/<raw_device> of=<temp_file> bs=4096 count=<datafile_size>
  ```

- If Oracle uses all the space in the logical volume, you have to expand the
  logical volume. Since the data will be skipped by one block (4 KB) and we
  need space to hold that extra block, use the System Management Interface
  Tool (SMIT) or the `chlv` command to increase the size of the logical volume, if
  needed.

- Restore data back to the raw device:

  Use the following command to restore the data into the raw device.

  ```
  $ dd if=<temp_file> of=/dev/<raw_device> bs=4096 seek=1 count=<datafile_size>
  ```

  Note: Use the information that you have from the section "Identifying all Files
  Related to a Database" on page 251 Replace `<raw_device>` with your raw
  device name. Replace `<temp_file>` with your temporary file or backup device
  name. And replace `<datafile_size>` with the size of the data file in 4 KB
  blocks. If you omit the count parameter, `dd` will copy the entire file.

Repeat this step for all of the raw devices in the database.

5.6.1.5 Changing Environment Settings

Now you have to plan which directory to install the new Oracle V7.1.6 into. Oracle
recommends creating a separate directory for each release. You can embed
version and release numbers in the path name.

For example:

If the old ORACLE_HOME is /u01/home/dba/oracle/product/7.0.15, the new
ORACLE_HOME should be /u01/home/dba/oracle/product/7.1.6.

Then change ORACLE_HOME and PATH environment variables to the new
directory where you plan to install the new release.

5.6.1.6 Installing Oracle V7.1.6

Start the Oracle installation process as usual, but do not create the new database object. The purpose of this step is to copy Oracle code from the distribution media to disk only.

- Copy the Installer from Oracle distribution media to disk.

  If you install from tape, use `cpio` to extract the Installer from the first block of the distribution tape.

  If you install from CD-ROM, mount the CD-ROM on a temporary directory, and call the Oracle shell script to copy the Installer to disk. For example:

  As root, enter the following command:

  ```
  # mount -o ro -v cdrfs /dev/cd0 /cdrom
  ```

  Then, as Oracle owner, enter the following command:

  ```
  $ mkdir <oracle_link>
  $ cd /cdrom/orainst
  $ ./start.sh
  ```

  where <oracle_link> is the name of a temporary directory that will be used by Oracle as a working area during Oracle installation. The start.sh script will ask for the name of that directory.

- As root user, run the rootpre.sh script to set up the environment and the kernel extensions. If you have an older version of kernel extensions already loaded, you have to reboot the machine to load the new version.

  ```
  # cd <oracle_link>
  # ./rootpre.sh
  ```

- Invoke the Installer:

  ```
  $ cd <oracle_link>/orainst
  $ ./orainst
  ```

  The Installer will ask for the ORACLE_HOME directory; enter the new directory for the Version 7.1.6.

- Select **Install/Upgrade/Patch Software Only** in the Install Actions screen.

```
 File  Edit  Options  Help
┌──────────────────── Install Actions ────────────────────┐
│                                                          │
│   Select the desired Installer action:                   │
│                                                          │
│                                                          │
│   ┌────────────────────────────────────────────────┐█   │
│   │  COMPLETE SOFTWARE/DATABASE FRESH INSTALL       │█   │
│   │  Install/Upgrade/Patch Software Only            │    │
│   │  Create New Database Objects                    │    │
│   │  Upgrade Existing Database Objects              │    │
│   │  Software/Database Maintenance                  │    │
│   │  V6 to ORACLE7 Migrate                          │    │
│   │  Patch Deinstallation                           │    │
│   │                                                 │    │
│   └────────────────────────────────────────────────┘    │
│                                                          │
│                                          (Select)        │
│                                                          │
│     (Help...)              (Back)          (Cancel)      │
│                                                          │
└──────────────────────────────────────────────────────────┘
```
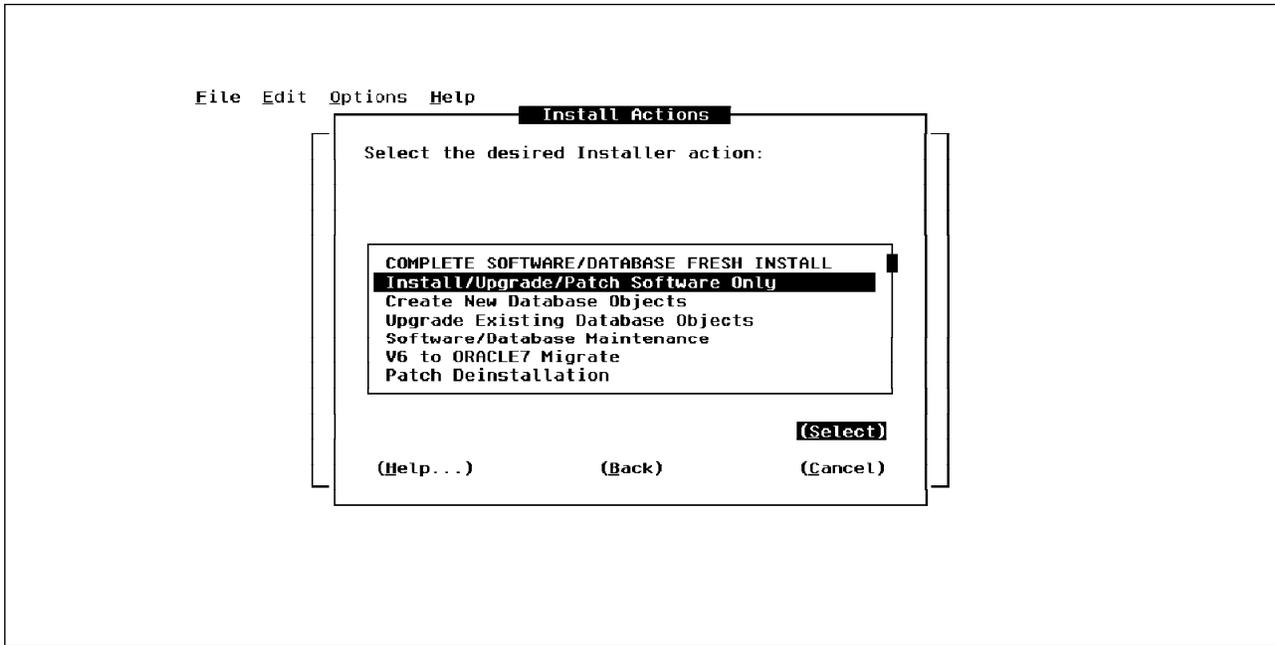
Figure 52. Oracle Install Actions Screen

- When the Installer shows available products on the screen, select all Oracle products that you want to install; press the **Install** button, and press **Enter** to proceed.

```
 File  Edit  Options  Help
┌──────────── Available Products ────────────┐ Products ──────────────┐
│                                            │                         │
│   Products available on /usr/oracle/olink  │  /usr/oracle/7.1.6      │
│                                            │                         │
│   ┌─────────────────────────────────────┐  │ ┌─────────────────────┐ │
│   │ ORACLE7 Distributed Database option 7.1.6.│ad Files 1.0.0.0.1 █│ │
│   │ ORACLE7 Parallel Query option 7.1.6.2.0   │2.0                 │ │
│   │ ORACLE7 Parallel Server Option 7.1.6.2.0 █│es and Utilities 7.1.│ │
│   │ ORACLE7 Server (RDBMS) 7.1.6.2.0          │0.0.4               │ │
│   │ ORACLE7 Symmetric Replication option 7.1.6│atabase option 7.1.6.│ │
│   └─────────────────────────────────────┘  │ └─────────────────────┘ │
│                                            │                         │
│  142M bytes selected in 11 products. 61M bytes                       │
│  remain on the destination volume.                                   │
│                                            │                         │
│  [Install...]                (From...)     │        (To...)          │
│                                            │                         │
├────────────────────────────────────────────┴─────────────────────────┤
│  Installs selected products.  [Space]-Invoke  [Return]-Invoke  [Tab]- │
│  Navigate  [K0]-Menu  [K3]-NextWindow  [PF1 K5]-MoveWindow            │
└───────────────────────────────────────────────────────────────────────┘
```

Figure 53. Oracle Products-List Screen

This operation copies Oracle code from the distribution media, but it does not make any change to the database.

5.6.1.7 Modifying Configuration Files

Copy the configuration files, initSID.ora and configSID.ora, to the new ORACLE_HOME/dbs directory. These files contain pointers to the database control files. The path name of the control files may contain a symbol **?** (question mark) that means it points to ORACLE_HOME directory. If there is a ? symbol, replace the symbol with the old ORACLE_HOME. If it already uses a full path name, you do not need to change it.

5.6.1.8 Upgrading the Database Objects

┌─ **Attention!** ───┐

This step will modify your database files. Make sure that you have backed up all the files related to the database.

└──┘

Follow these steps to upgrade your database objects:

- As Oracle user, start the Installer.

- When the Installer prompts for ORACLE_HOME, enter the new V7 ORACLE_HOME.

- Select the **Upgrade Existing Database Objects** option in the Installer Actions menu.

- Enter the ORACLE_SID for the database to be upgraded.

- The Installer will show a list of products. Select the Oracle RDBMS server and the other Oracle products whose objects you want to upgrade to the new version. Then press the **Install** button, and press **Enter** to perform the upgrade.

The Installer will upgrade the database objects (tables used by Oracle products) to support Oracle V7.1.6. For example, if you select to upgrade SQL*Forms, the Installer will update the tables used by SQL*Forms and change the name of the forms created by end-users to capital letters. This step may take some time.

5.6.1.9 Updating Configuration in /etc

Modify the /etc/oratab file to update the new ORACLE_HOME directory for this database. For example, if you upgrade the PROD database to V7.1.6, the oratab file should look like this:

PROD**:/u01/home/dba/oracle/product/7.1.6:**Y

If SQL*Net is installed, the SQL*Net configuration file should also be updated. Look for the parameter ORACLE_HOME in /etc/listener.ora, and replace the old ORACLE_HOME with the new V7 ORACLE_HOME. For example:

```
SID_LIT_LISTENER =
  (SID_LIST =
    (SID_DESC =
      (SID_NAME = PROD)
      (ORACLE_HOME = /u01/home/dba/oracle/product/7.1.6)
    )
  )
```

5.6.1.10 Testing

Now you can start the database and test your application. Make sure that ORACLE_HOME and PATH point to the new ORACLE_HOME.

You may have to change your application to support the new release in order to exploit the new features of that release. Please consult your Oracle support for details.

If you have more than one database, repeat steps 1.5.1.2 to 1.5.1.4 and 1.5.1.7 to 1.5.1.10 to upgrade another database.

5.6.2 Upgrading Oracle V6 to V7.1.6 on AIX V3.2.5

The database structure in V6 is different from the structure in V7. In order to simplify the migration procedure, Oracle provides the Migration Utility to change the database structure and maintain existing data in the database. If you have Oracle V6 installed, you can either use the Migration Utility or the Export/Import utility to migrate your database.

If you have Oracle V6.0.27 or later, use the following method to migrate Oracle to V7.1.6:

- Document the system.
- Install the Migration Utility.
- Shut down the old database.
- Back up the old database.
- Modify raw devices.
- Run the Migration Utility.
- Install the new Oracle V7.
- Clean up the database files.
- Upgrade the database objects.
- Create V6 compatibility views.

5.6.2.1 Documenting the System

Use the same procedure described in section 5.6.1.1, "Documenting the System" on page 250 to collect information about the databases in your system. However, in Oracle V6, you may not have the configSID.ora file.

5.6.2.2 Installing the Migration Utility

The purpose of this step is to use the Oracle Installer to copy the Migration Utility into your V6 ORACLE_HOME directory. When finished, you will see the migrate program in ORACLE_HOME/bin. This program will be used later.

- Copy the Oracle Installer from Oracle V7.1.6 distribution to your disk.
- As root user, run the rootpre.sh script.
- As Oracle owner, start the Installer.
- The Installer will prompt for the ORACLE_HOME directory; enter the V6 ORACLE_HOME directory.
- Select the **V6 to ORACLE7 Migrate** option in the Install Actions screen.
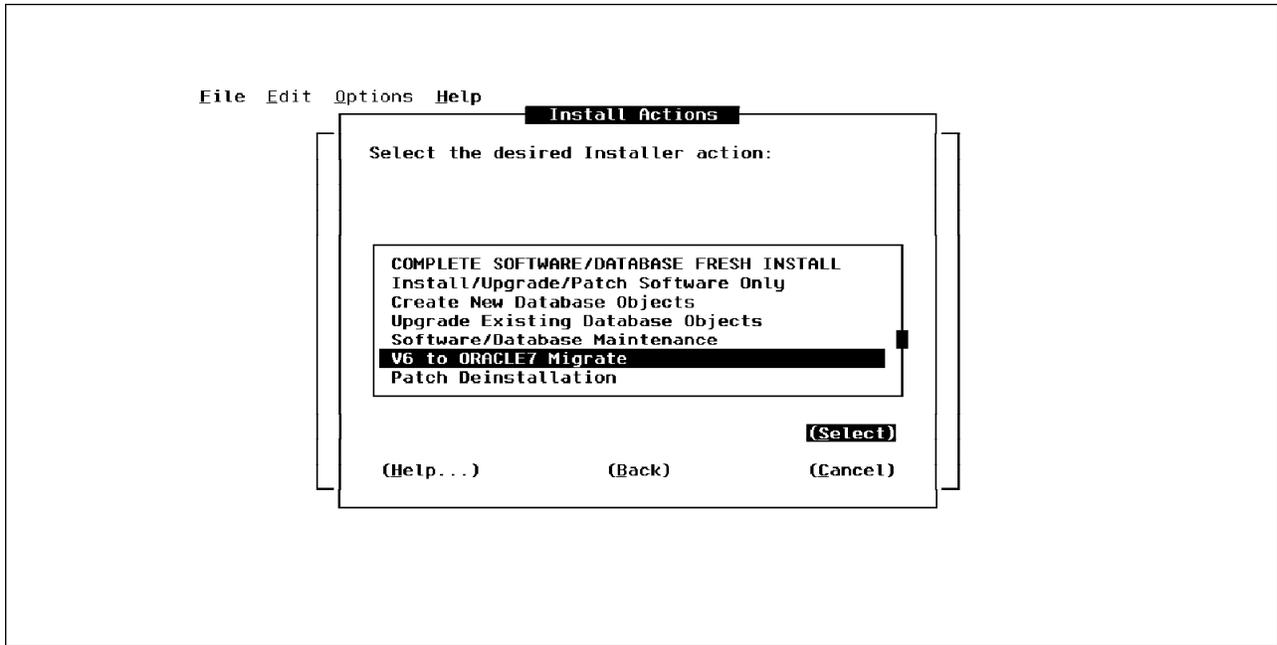
Figure 54. Oracle Install Actions Screen

- Select the **Install Migration Utility** option from the Migration screen.
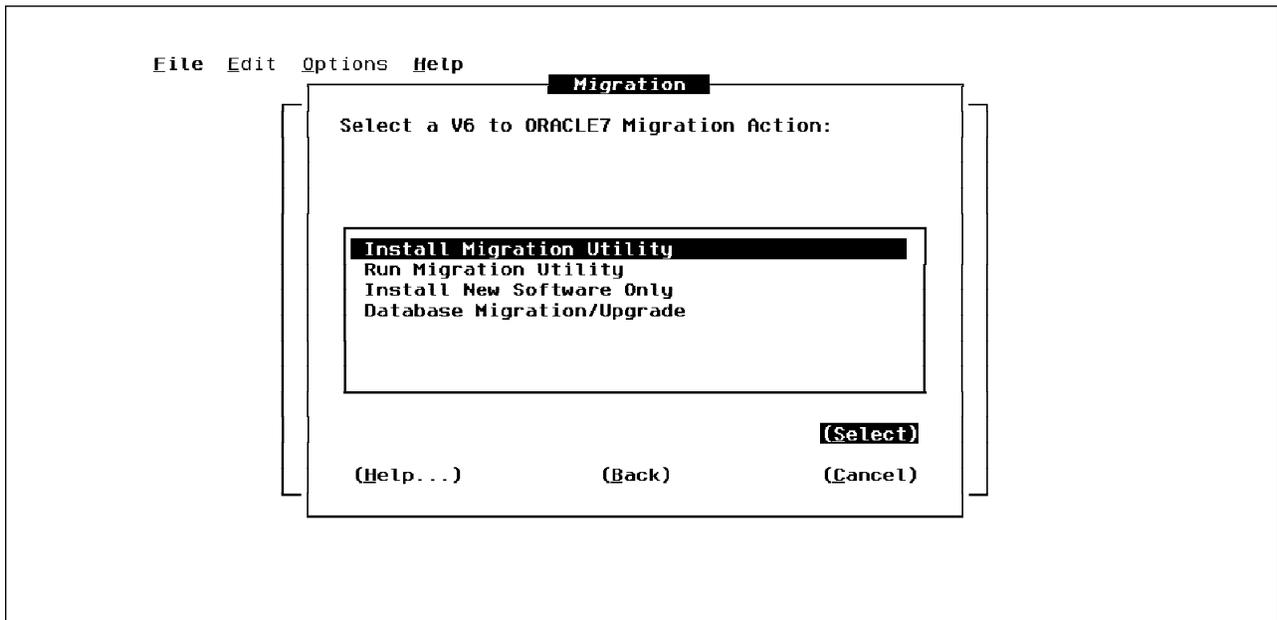
Figure 55. Oracle Migration Screen

- The Installer will show all products available on the media, select only the **V6 to V7 Migration Utility**. Press the **Install** button, and press **Enter** to start.

```
 File  Edit  Options  Help
        ┌──────────Available Products──────────┐   ┌─Products──────────
        │      Products available on /usr/oracle/olink   │  /usr/oracle/7.1.6
        │                                                │
        │   ┌────────────────────────────────────────┐  │  ┌──────────────────┐■
        │   │  SQL*Report 1.1.10.3.1                  │  │  │ad Files 1.0.0.0.1│
        │   │  SQL*ReportWriter 1.1.14.15.1           │  │  │2.0               │
        │   │  TCP/IP Protocol Adapter (V2) 2.1.6.1.0 │  │  │es and Utilities 7.1.│
        │   │  ToolKit 2 Extra Printer Files 2.0.10.22.0│ │  │0.0.4             │
        │   │  V6 to V7 Migration Utility. 7.1.6.2.0  │■ │  │atabase option 7.1.6.│
        │   └────────────────────────────────────────┘  │  └──────────────────┘
        │                                                │
        │   1989K bytes selected in 1 product. 61M bytes │
        │   remain on the destination volume.            │
        │                                                │
        │    ▐Install...▌               (From...)        │           (To...)
        │                                                │
        └────────────────────────────────────────────────┘
        ┌────────────────────────────────────────────────┐
        │   Migration Installation                        │
        │                                                 │
        └─────────────────────────────────────────────────┘
```

Figure 56. Oracle Products-List Screen

- When finished, exit from the Installer.

5.6.2.3 Shutting Down the Old Database

In the SQL*DBA utility for Version 6, you do not need to use the command `connect` before shutting down the database. Use the following command to shut down the V6 database:

```
$ export ORACLE_SID=<database_name>
$ sqldba
SQLDBA> shutdown
SQLDBA> exit
```

5.6.2.4 Backing Up the Old Database

Back up all files related to this database to the backup media. Use the same procedure described in section 5.6.1.3, "Backing up an Old Database" on page 253.

5.6.2.5 Modifying Raw Devices

If you use a raw device to store the database file, use the same procedure in section 5.6.1.4, "Modifying Raw Devices" on page 254 to modify the raw device before it can be used with Oracle V7.1.6.

5.6.2.6 Running the Migration Utility

In this step, we will use the Oracle Installer to invoke the migrate program that was installed in a previous step. The migrate program will modify the structure of your Oracle V6 database and create the convSID.dbf file that will be used in the next step.

- Start the Installer.

- Select the **V6 to ORACLE7 Migrate** option in the Install Actions screen.

- Select the **Run Migration Utility** option from the Migration screen.

- The Installer will prompt for ORACLE_SID; enter your database name (SID).

- The Installer will show all products available on the media, select only the **V6 to V7 Migration Utility** and press the **Install** button, and press `Enter` to start.

When finished, the Installer will show instructions that you should perform before proceeding to the next step. These instructions will be discussed in the next step. Verify that a file convSID.dbf has been created under the ORACLE_HOME/dbs directory.

5.6.2.7 Installing Oracle V7.1.6
This step will only copy the Oracle code from the distribution media to disk. You should create a new directory for V7.1.6, and use the Installer to copy files from the distribution media to disk.

- Start the Installer.

- Select the **V6 to ORACLE7 Migrate** option in the Install Actions screen.

- Select the **Install New Software Only** option from the Migration screen.

- The Installer will show all products available on the media; select all the Oracle products that you want to install.

When finished, exit the Installer.

5.6.2.8 Cleaning Up the Database Files
Oracle has added several features to V7 that make some parameters in the database configuration files become obsolete. There are also some new parameters that you have to add to the configuration files.

- Copy initSID.ora file from the V6 ORACLE_HOME/dbs to the new V7 ORACLE_HOME/dbs.

- Edit initSID.ora file to remove obsolete parameters, and add the new V7 parameters. See *Oracle7 Server Migration Guide* for details.

- Copy convSID.dbf that is generated by the migrate program from V6 ORACLE_HOME/dbs to the new V7 ORACLE_HOME/dbs.

- Remove the database control files or move them to another directory. The new control files for V7 will be recreated by Oracle.

5.6.2.9 Upgrading Products
This step creates new control files and changes the database structure to support Oracle V7. After this step, you can use V7 applications to access data in the database and test the new environment.

- Start the Installer.

- Select the **V6 to ORACLE7 Migrate** option in the Install Actions screen.

- Select the **Database Migration/Upgrade** option from the Migration screen.

- The Installer will show all products available on the media; select the new products that you want to install.

This step will take some time.

5.6.2.10 Creating V6 Compatibility Views

The data dictionaries of Oracle V6 and V7 are different. If you have not migrated
your applications and the old application still requires the V6-compatible data
dictionary, you can create V6 data dictionary views by using an SQL script called
`catalog6.sql`. As Oracle owner, issue the following command:

```
$ sqldba lmode=y
SQLDBA> @$ORACLE_HOME/rdbms/admin/catalog6.sql
SQLDBA> exit
```

If you have more than one database in the machine, repeat steps 1.5.2.3 to 1.5.2.5
and 1.5.2.7 to 1.5.2.9 to upgrade another database.

5.6.3 Migrating Oracle V6 or V7 to V7.1.6 Using Export/Import Utilities

In some early releases of Oracle, it is recommended to use the Export/Import
utilities instead of the Migration Utility or the Installer. See 5.3.3, "Oracle Upgrade
Paths" on page 237 for the specific versions of Oracle that require this method.

This method will move the whole data in a database out to the backup media and
then bring the data back into the new database again. This method consists of
more steps and has a higher risk than the two methods described earlier in this
section, especially when you have to remove the old database and restore the data
back in again. However, it provides more flexibility to move data to a different
system or to change the database configuration. If you have enough disk space,
you can keep both new and old versions of the database at the same time.

The following list contains some major steps used to perform a migration using
Export/Import utilities.

5.6.3.1 Documenting your System

Use the procedure described in section 5.6.1.1, "Documenting the System" on
page 250 to record information about the existing database in your system.

5.6.3.2 Exporting the Full Database

This step will shutdown the database to stop user activity, and restart the database
in a special mode that will allow only the database administrator to use the
database. The Export utility is used to convert all data in the database to a large
sequential file that will be used to import data from in a later step.

Log in as Oracle owner, and use the following command:

- For Oracle Version 6:

```
$ export ORACLE_SID=<database_name>
$ sqldba
SQLDBA> shutdown normal
SQLDBA> startup dba
SQLDBA> exit
$ exp full=y grants=y constraints=y userid=system/<manager> file=<export_file>
```

- For Oracle Version 7:

```
$ export ORACLE_SID=<database_name>
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> shutdown normal
SQLDBA> startup restrict
```

```
SQLDBA> exit
$ exp full=y userid=system/<manager> file=<export_file>
```

Note: Replace `<manager>` with your system account password. The `<export_file>` can be a temporary file or a backup device name, for example: /dev/rmt0.

5.6.3.3 Shutting Down and Backing up the Old Database

Follow the instructions in section 5.6.1.3, "Backing up an Old Database" on page 253 to back up the database.

5.6.3.4 Removing the Old Database (optional)

If your system has enough free space to keep both the old and the new database you do not need to remove the old database files. In this case, the recovery process is easier. You will just have to change the configuration, and restart the database.

If there is not enough space in the system, you can remove the old data files to free some space. Remove data files that you have already backed up in the previous step by using the `rm` command.

5.6.3.5 Installing the Oracle V7.1.6 and Creating the New Database

Start the Oracle installation process. Follow these steps to install the new release of Oracle.

- Copy the Installer from the Oracle distribution media to disk.

- As root user, run the `rootpre.sh` script to set up the environment and the kernel extensions. If you have an older version of the kernel extensions already loaded, you have to reboot the machine to load the new version.

- Invoke the Installer to install Oracle:

  Since the new database will be created from scratch, you can have the Oracle Installer create the new database for you, or you can create it later using SQL commands. It is easir to have the Installer create the database, but the Installer will use default parameters to create it. If you want to specify initial parameters, such as the db_block_size, you have to create the database manually by using SQL commands.

  – To have the Installer install Oracle and create the database:

    Select the **COMPLETE SOFTWARE/DATABASE FRESH INSTALL** option in the Install Actions screen. The Installer will copy the Oracle code from the distribution media to the ORACLE_HOME directory, ask for the database name or SID, data file name and size and then create the database for you.

  – To install Oracle software only:

    Select the **Install/Upgrade/Patch Software Only** option in the Install Actions screen. This operation will copy Oracle code from the distribution media to ORACLE_HOME only.

    See *Oracle7 for AIX Installation and Configuration Guide* for details on how to create the database using SQL commands.

5.6.3.6 Importing the Database

Now you should have Oracle V7.1.6 installed with the new database up and running. Log in as Oracle owner, and use the following step to bring the data from the export file back to the new database.

```
$ imp full=y ignore=y userid=system/<manager> file=<export_file>
```

Note: Replace `<manager>` with your system account password and `<export_file>` with the export file that you have created in step 1.5.3.2.

Tips for Using the Import Utility

The Import utility uses the rollback segment to hold data for recovery. If the table is very large, the rollback segment may be insufficient, and the Import utility will report an error message like the following:

```
IMP-00003: ORACLE error 1562 encountered
ORA-01562: failed to extend rollback segment (id = 2)
ORA-01547: failed to allocate extent of size 35 in tablespace 'RBS'
IMP-00028: partial import of previous table rolled back: 18011 rows rolled back
```

If you have this problem, you can create a larger tablespace, and then create a rollback segment on that new tablespace. Or you can use the option `commit=y` when you start the `imp` command. This option forces the Import utility to commit transactions regularly and frees up space in the rollback segment.

5.6.3.7 Updating Configuration Files

Update the /etc/oratab and /etc/listener.ora files. Add a new entry for the new database with the new ORACLE_HOME, and remove the old entry of the old database.

5.6.3.8 Testing

Now you can test your application. Make sure to set ORACLE_HOME and PATH variables to the new ORACLE_HOME.

If you have more than one database, repeat steps 1.5.3.2 to 1.5.3.4 and 1.5.3.6 to 1.5.3.8 to migrate to another database.

5.6.4 Recovering from an Oracle Migration Failure

Once the database has been upgraded to the new version. The only way to go back to the old version is to restore the database. If new transactions have been made to the new database, you will lose these changes.

Follow these guidelines to restore the old version of the database.

5.6.4.1 Shutting Down the New Release

As Oracle owner, issue the following command to stop the database:

```
$ export ORACLE_SID=<database_name>
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> shutdown
SQLDBA> exit
```

5.6.4.2 Restoring Kernel Extensions

The kernel extensions of the old version are replaced by the new versions during the Oracle migration. You should restore the old kernel extensions back to the /etc directory and load them into the AIX kernel. You can find the kernel extensions under the Oracle directory:

- For Oracle V6: under $ORACLE_HOME/rdbms/lib

- For Oracle V7: under the directory where you copied the Oracle Installer into.

Copy the following files to the /etc directory, and reboot the machine to load the old kernel extensions to the system.

Table 25. Oracle Kernel Extension Files

File Name	Description
pw-syscall	Post/wait kernel extensions
adspc	Protect shared memory kernel extensions
adspcinit	Program to initialize protect shared memory kernel extensions
loadext	Program to load kernel extensions into the AIX kernel

5.6.4.3 Restoring the Database

Use AIX utilities to restore all the database files that you have backed up before the migration.

Verify that the following files are restored completely:

- Configuration files

- Control files

- Data and log files

5.6.4.4 Changing Configuration Files

Update the ORACLE_HOME and PATH environment variables back to the directory where you installed the old version of Oracle. Also update /etc/oratab and /etc/listener.ora by changing ORACLE_HOME in these two files back to ORACLE_HOME of the old version.

5.6.4.5 Restart the Database

Restart your database, and check that all data files have been restored and your application works properly.

5.7 Migrating an AIX V3.2.5 System with Oracle V7.1.6

This section describes how to migrate an AIX V3.2.5 system with Oracle V7.1.6 to AIX Version 4.1.

5.7.1 Documenting the System

Use the same procedure described in section 5.6.1.1, "Documenting the System" on page 250 to collect information about the databases in your system.

5.7.2 Backing Up your System

In order to protect from a migration failure, the current working system must be backed up. The raw devices backup and AIX system backup must be performed. The other configuration files may be backed up if needed.

5.7.2.1 Backing Up Databases and Raw Devices

Use the same procedure described in section 5.6.1.3, "Backing up an Old Database" on page 253 to backup the databases. You must back up all data files in raw devices manually. For the data files in normal UNIX filesystem, they will be included in the AIX system backup in the next step. However, you may backup the whole database again to protect the data from media error or other failure.

5.7.2.2 AIX System Backup

Use SMIT or the mksysb command to back up AIX files to a backup media. Make sure that you have all file systems in rootvg mounted. Refer to Chapter 1, "Migrating AIX from V3.2.5 to V4.1" on page 3 in this redbook for more information. You can also refer to InfoExplorer for more details.

5.7.2.3 Backing up Other Configuration Files

Table 26 shows Oracle files installed in the /etc and /usr directories. If you plan to use the AIX Migration Install, these files will be preserved, and you do not need to do anything with these files after the migration. For an AIX Preservation Install or a New or Complete Overwrite Install, the content of / (root) and /usr will be removed; so you have to back up these files prior to the migration, and restore them into /etc and /usr after the AIX migration.

Table 26 (Page 1 of 2). Oracle Files in AIX Directories	
File Name	**Description**
/etc/adspc	Protected, shared memory kernel extensions.
/etc/adspcinit	Program to initialize protected, shared memory kernel extensions.
/etc/pw-syscall	Post/Wait kernel extensions.
/etc/loadext	Program to load kernel extensions.
/etc/oratab	This table contains a list of Oracle database names and directories.
/etc/listener.ora	SQL*Net configuration. It contains protocol, service port number and database name on the server. This is used by the listener process on the server machine.
/etc/tnsnames.ora	SQL*Net configuration. It contains the remote database name and network parameters used by applications to connect to the remote database.
/etc/intchg.ora	SQL*Net configuration.
/etc/tnsnet.ora	SQL*Net configuration.
/etc/tnsnav.ora	SQL*Net configuration.
/etc/sqlnet.ora	SQL*Net configuration. It contains diagnostic and other optional parameters.

Table 26 (Page 2 of 2). Oracle Files in AIX Directories	
File Name	**Description**
/etc/sqlnet	SQL*Net V1 configuration. It contains database connection strings.
/usr/lbin/coraenv	Shell script to set Oracle environment (C shell).
/usr/lbin/oraenv	Shell script to set Oracle environment (Bourne shell).
/usr/lbin/dbhome	Shell script called by oraenv to determine Oracle home directory.

During the Oracle installation, some AIX configuration files will be updated.
Table 27 contains AIX files that are updated by Oracle and the effect of the AIX
migration on this files.

The inittab file will be updated by Oracle in order to load the kernel extensions into
the AIX kernel during IPL. There are two of them, protected shared memory
(adspc) and post/wait (pw-syscall).

Attention!

The Oracle kernel extensions for AIX V3.2 will NOT work with AIX V4.1. So,
you have to get the appropriate files for AIX V4.1.

Some system administrators may update /etc/inittab or /etc/rc to automatically start
the database. After the migration, both files will be replaced. You will have to
update them manually. See section 5.7.4.2, "Automatic Database Startup" on
page 269 for a procedure to update the inittab and rc files after the migration.

If SQL*Net is used or if the database is configured in a client/server environment,
the /etc/services file will be updated to specify the port number for the Oracle
listener. After the migration, this file will be merged with AIX V4.1. The new
service numbers will be added into this file. If the service number already exists,
the new service from AIX V4.1 will be commented out. You do not need to do
anything with this file.

Note: Recommended port numbers are 1525 for SQL*Net TCP/IP V1 and 1521
for SQL*Net V2.

All of these files will be included in the AIX system backup tape. However, you
may want to back up these files separately for reference or for protection from
failure.

Table 27. AIX Configuration Files Modified by Oracle		
File Name	**Description**	**After AIX Migration**
/etc/inittab	AIX initialization: loads Oracle kernel extensions	Replaced by AIX V4.1
/etc/services	Contains service port number for SQL*Net	Preserved
/etc/rc	AIX initialization	Replaced by AIX V4.1

5.7.3 Migrating AIX

Stop!

Make sure that you have done all the necessary backups before you proceed.
After you have installed AIX V4.1, the only way to go back is to restore the
system.

Before you start the AIX migration, read carefully Chapter 1, "Migrating AIX from
V3.2.5 to V4.1" on page 3 in this redbook, and then perform the following steps:

- Shut down all the databases.

- Shut down the machine normally.

Migrate AIX using the Migration Install method.

The Oracle Installer calls AIX utilities, such as make, ld and ar to perform linking.
Make sure that you have the necessary filesets installed after the migration.
Usually, if you use the AIX Migration Install method and had the basic application
development Optional Program Products (OPP) installed in AIX V3.2.5, the
following packages will be installed automatically during the migration:

- bos.adt.base: for `make` and `as` (assembler) commands

- bos.rte.bind_cmds: for ld, ar and other library utilities

5.7.4 Post Migration Steps

5.7.4.1 Oracle Kernel Extensions
In AIX V3.2, during the Oracle installation, the shell script rootpre.sh must be
excuted by root in order to properly set up the environment for Oracle. The
rootpre.sh script copies kernel extension files from the Oracle directory to the /etc
directory and updates the /etc/inittab file to automatically load the kernel extensions
during the AIX startup.

This is the entry in the /etc/inittab file:

```
orapw:2:wait:/etc/loadext /etc/pw-syscall3.2
orakx:2:wait:/etc/loadext /etc/adspc
orakx2:2:wait:/etc/adspcinit
```

In AIX V4.1, the old kernel extensions will not be able to work with the AIX V4.1
kernel. If you try to load the old kernel extensions you will get an error message
like this:

```
SYS_SINGLELOAD: Exec format error
kmid: 0 (0x0)
path: '/etc/pw-syscall3.2'
libpath: ''
```

When you migrate to AIX V4.1, the /etc/inittab file will be replaced by the new one
from the AIX V4.1 distribution. You have to re-run the rootpre.sh script to copy the
appropriate version of the kernel extensions (pw-syscall and adspc). The
rootpre.sh script also updates /etc/inittab in order to load the kernel extensions
during AIX startup.

Log in as root, and run the following command:

```
# cd $ORACLE_HOME/orainst
```

Or if you install Oracle from a CD-ROM:

```
# cd ORACLE_LINK_DIR/orainst
```

Then run the shell script:

```
# ./rootpre.sh
```

5.7.4.2 Automatic Database Startup

In AIX V3.2, the database admistrator may modify the AIX configuration to automatically start up the database during IPL. Usually, a startup command is added to /etc/inittab or /etc/rc. After the migration, both /etc/inittab and /etc/rc will be replaced by AIX V4.1, and the old files will be moved to /tmp/bos/etc/inittab and /tmp/bos/etc/rc. We recommend making a new entry in the /etc/inittab and not modifying the /etc/rc. To update /etc/inittab, use the vi editor or use the `mkitab` command.

For example:

```
# mkitab "oradb:2:once:su - oracle -c dbstart"
# mkitab "oranet:2:once:su - oracle -c lsnrctl start"
```

The first line will start all databases that have the automatic start flag set to yes (Y) in /etc/oratab. The second line will start the SQL*Net V2 listener process.

5.7.4.3 Raw Devices

If data files are stored in raw devices, these raw device files must be owned by Oracle, must belong to the Oracle administrator group and must have permissions set to 600 in order to protect these files from unauthorized access.

This is an example of raw device files:

```
crw-------   1 oracle   dba        10, 14 Jul 14 10:21 /dev/roralog1
crw-------   1 oracle   dba        10, 15 Jul 14 10:21 /dev/roralog2
crw-------   1 oracle   dba        10, 16 Jul 14 10:21 /dev/rorarbs1
crw-------   1 oracle   dba        10, 17 Jul 14 10:21 /dev/rorasys1
crw-------   1 oracle   dba        10, 18 Jul 14 10:21 /dev/rorausr1
```

During the AIX V4.1 migration, the /dev directory will be recreated, and all the files under /dev will be reconfigured. So, the owner of all logical volumes will be reset to root, the group to system and permissions will be set to 660. If you try to start the database, Oracle will display an error message like this one:

```
SQLDBA> connect internal
Connected.
SQLDBA> startup
ORACLE instance started.
Database mounted.
ORA-01157: cannot identify data file 1 - file not found
ORA-01110: data file 1: '/dev/rorasys1'
Attempting to dismount database........Database dismounted.
Attempting to shutdown instance........ORACLE instance shut down.
SQLDBA>
```

Owners and permissions of data files must be changed back to Oracle. Then the database can be started.

To change the ownership and permission of the data files, log in as root, and run the following command:

```
# cd /dev
# chown oracle.dba <datafile_name>
# chmod 600 <datafile_name>
```

Where `oracle` is the user name of the Oracle owner, `dba` the database administrator group on your machine and `<datafile_name>` the name of your data file in /dev.

5.7.4.4 LFT versus HFT

AIX V4.1 does not support High Function Terminal (HFT), but it provides Low Function Terminal (LFT) support that has very similar characteristics. For some Oracle applications that may not recognize the LFT terminal, use the HFT terminal-type instead.

For sqldba and orainst, do the following:

```
$ TERM=hft ; export TERM
$ sqldba
```

For sqlforms30 , sqlmenu50 or oraterm, specify the following option:

```
$ sqlforms30 -c hft:hft
```

5.7.4.5 Problem with Direct Attached TTYs

There is a problem with Oracle running on TTYs attached directly to an IBM RISC System/6000. The problem affects Oracle applications that utilize full-screen features like SQL*Forms, SQL*Menu and SQL*Report.

At the time this redbook was written, we found the problem in the following situations:

 1. When exiting the application:

    A user can start an Oracle application, like sqlforms30, oraterm and so on, and use all functions as usual. But, after exiting from the application, the terminal will display garbage and will no longer accept any command.

 2. When an application calls another subprogram:

    This problem also occurs when the application calls a subprogram or another application. For example, SQL*Menu may call SQL*Report to run a report. When SQL*Menu starts SQL*Report and handles the screen to SQL*Report, the characters will be displayed as garbage and cause a core dump.

This problem comes from a wrong terminal setting. One of the TTY characteristics, character size, was set to 5 (cs5) instead of 8 (cs8). This problem will not affect the graphic console (LFT), a remote terminal (pts) or some other terminal types that ignore the character-size setting.

To temporarily bypass this problem, you can either telnet to the local machine in order to create a pseudo-terminal that will ignore the wrong setting, or change the terminal characteristic to prevent the problem by issuing the following command

```
stty -opost
```

However, this option will produce some side-effects. When you press the Enter key, the TTY will not generate a carriage return code. So, after exiting from the application, you have to reset it back to normal.

This example shows how to use telnet to bypass the TTY problem:

```
$ telnet localhost
$ sqlforms30
$ exit
```

This example use `stty`:

```
$ stty -opost
$ sqlforms30
$ stty opost
```

5.7.4.6 IBM-850 Code-Page Changed to ISO-8859

If your machine has the language environment set to En_US and uses the IBM-850 code page in AIX V3.2, after the migration, the language environment will be changed to en_US and the code page changed to ISO-8859. This new code page may cause a minor problem with some character-based applications, especially when the application displays a non-alphanumeric character. For example, the border of the text window or the pull-down menu are displayed with the wrong font. This problem affects only the graphic console (LFT) and aixterms.

To resolve this problem:

- Use SMIT to change the primary language to IBM-850 (English United States [En_US])

  If there is no fileset for this language installed on the machine, it will automatically install it from the installation device (tape or CD-ROM).

- Reboot the machine if you have the problem on the graphic console (LFT).

- Or restart X-Windows if it is an aixterm.

5.7.4.7 Asynchronous I/O

Oracle uses asynchronous I/O to transfer data between the Oracle daemon process and data files. During Oracle installation, user root must run the rootpre.sh script to set up the device. The rootpre.sh script configures the aio0 device to become available on every system restart.

After the AIX migration, the characteristics of the aio0 device are preserved. So, it is ready to be used by Oracle.

If you install AIX with the New and Overwrite Install method, the aio0 device will not be configured. Its status will be defined. However, when you run rootpre.sh again, the shell script will configure the aio0 device for you.

If you find an error message like the following, it means that the device aio0 is still in the defined state. Use SMIT or the `chdev` command to change the aio0 device to make it available.

```
0509-037 System error - error data is: oracle
0509-023 Symbol kaio_rdwr in /usr/lib/libc.a is not defined.
0509-023 Symbol listio in /usr/lib/libc.a is not defined.
0509-023 Symbol acancel in /usr/lib/libc.a is not defined.
0509-023 Symbol iosuspend in /usr/lib/libc.a is not defined.
0509-022 Cannot load library libc.a[aio.o].
```

To improve the disk I/O performance, Oracle recommends storing data files in raw devices and putting the following parameter in the database configuration file (initSID.ora):

```
use_async_io = true
```

5.7.5 Testing the New Environment

Use the testing plan that you have developed in the planning section. Run the tests, and compare the results with the one on AIX V3.2 system.

5.7.6 Recovering AIX V3.2.5

If you find a problem in an AIX V4.1 system and need to go back to AIX V3.2, you have to reinstall the system. Follow these steps to restore your system.

5.7.6.1 Restoring AIX

Reinstall AIX by booting the system with a system backup tape. When AIX shows the installation menu on the screen, select option 2 **Install a system that was created with the SMIT "Backup The System"**.

This step will restore all AIX V3.2.5 files, including Oracle kernel extensions for V3.2.5 and Oracle configuration files in /etc and /usr. So, after rebooting the machine, it will load the proper kernel extensions into AIX, and the Oracle configuration files are restored to the same state as before the migration.

5.7.6.2 Restoring Raw Devices

Use the following guidelines to restore raw devices:

- Create a logical volume by using SMIT or the `crlv` command

  Using information that you have from the documenting step, create logical volumes with the same name and size.

  Note that a logical volume has two file names associated with it. The first one has the same name as the logical volume, and it is a block device; the second one has a character "r" preceeding the name, and it is a character device.

  For example, if you create a logical volume named "oradata", AIX will create two files in the /dev directory like the following:

  ```
  brw-rw----   1 root     system    10, 12 Jul 17 15:33 /dev/oradata
  crw-rw----   1 root     system    10, 12 Jul 17 15:33 /dev/roradata
  ```

  Usually, Oracle uses character device or /dev/roradata in this example.

- Change owner and permissions:

  A newly created logical volume has a default owner and permission. Use the following command to give the Oracle owner an authorized access to this file.

  As root user, issue the commands:

```
# cd /dev
# chown oracle.dba <datafile_name>
# chmod 600 <datafile_name>
```

- Restore data into the logical volume:

  Use the `dd` command to restore data from the backup media back to the logical volume.

5.7.6.3 Post-Restore steps

These are some post-restore steps:

Device renaming:

During an AIX system restore, since the device configuration database is rebuilt and merged with the ODM from a backup image, some device names may be changed. See Chapter 1, "Migrating AIX from V3.2.5 to V4.1" on page 3 in this document for more details.

Non-rootvg information:

Information in non-rootvg volume groups are preserved after you migrate and restore the AIX system. However, if you created a filesystem or a new logical volume that utilizes AIX V4.1 features (for example, disk striping, compressed filesystem), you will NOT be able to import that volume group on AIX V3.2 again. You have to recreate the volume group, and restore data from the backup media.

Raw devices in non-rootvg volume groups:

If you have a raw device in a non-rootvg volume group, its data will be preserved during the migration or the mksysb restoration, but the owner of the file in /dev will be reset to root. You have to change the owner of the file back to the Oracle owner by using the same command as described in 5.7.6.2, "Restoring Raw Devices" on page 272.

5.8 Migrating a Client/Server Environment

The purpose of this section is to describe the structure of the Oracle product in a client/server environment and to provide general guidance on how to migrate several machines to AIX V4.1.

5.8.1 Oracle Client/Server Environment Overview

In a centralized or a stand-alone environment, an application communicates with the server on the same machine using normal UNIX Inter Process Communications (IPC). The application specifies the database name or SID via a variable called ORACLE_SID to select which database server to talk to.

In a client/server environment, the client application and the database server are located on separate machines. They require a product called SQL*Net on both the client and the server machine in order to communicate with each other. The SQL*Net provides transparent access from the client application to the database server as if they were on the same machine. The application must specify a database name, the location of the server and network parameters to SQL*Net to make a proper connection. This information is called the *connection string* in SQL*Net V1 and the *connection descriptor* in SQL*Net V2.

There are two versions of SQL*Net, V1 and V2. SQL*Net V2 has several enhancements over V1. For example, V2:

- Supports a multiprotocol network

- Supports a multithreaded server in Oracle7

- Provides configuration tools for ease of use

- Has a new connection method

Oracle still provides support for SQL*Net V1, but SQL*Net V2 should be used for a new installation. Sites with existing SQL*Net V1 should consider migrating to SQL*Net V2 in the future.

The limitation for SQL*Net V1 and V2 is that SQL*Net V1 cannot communicate with SQL*Net V2. You can have both SQL*Net V1 and V2 running on the same machine concurrently. The SQL*Net V1 client will connect to only the SQL*Net V1 server, and the SQL*Net V2 client will connect to only the SQL*Net V2 server.

Figure 57 on page 275 shows a sample configuration of a client/server environment. On the server machine, you can have both versions of SQL*Net. You have to specify different network parameters for each version. For TCP/IP, each version must have a different port number. Recommended port numbers are 1525 for SQL*Net V1 and 1521 for SQL*Net V2. On the client side, SQL*Net V1 and V2 can be on the same machine or on different machines. If there are two versions on the same machine, Oracle will use the database connection string to determine which version of SQL*Net to use. If the user specifies the connection string in SQL*Net V1 format, the Oracle application will use SQL*Net V1. If the user specifies the connection descriptor in SQL*Net V2 format, the Oracle application will use SQL*Net V2.

5.8.2 Migration Method for a Client/Server Environment

As we discussed in the planning section, the new version of Oracle server can support most of the applications of the older version. So, the server machine should be migrated first.

The next step is to upgrade end-user applications and Oracle tools on client machines. If the current application uses the latest version of Oracle that already supports AIX V4.1, you can skip this step.

For example, if the client machine has SQL*Forms Version 3.0.16.12.9 or later installed, it does not need to be upgraded. But if you have an older version, you have to upgrade SQL*Forms.

If you use SQL*Net V1, you can choose to upgrade the SQL*Net to V2 or keep using the V1. Upgrading SQL*Net may require application modifications. If the programmer put a hard-coded database connection string in the application source code, you have to modify the connection string to the new format. As shown in Figure 57 on page 275, you can use SQL*Net V2 for some applications and use SQL*Net V1 for the others, until all the applications have been fully tested on SQL*Net V2; then you can remove the SQL*Net V1.

Once all the machines have the new version of Oracle that supports both AIX V3.2.5 and V4.1.3, you should test all applications on AIX V3.2.5 until you are sure that everything works correctly. Then you can migrate AIX to V4.1.3.
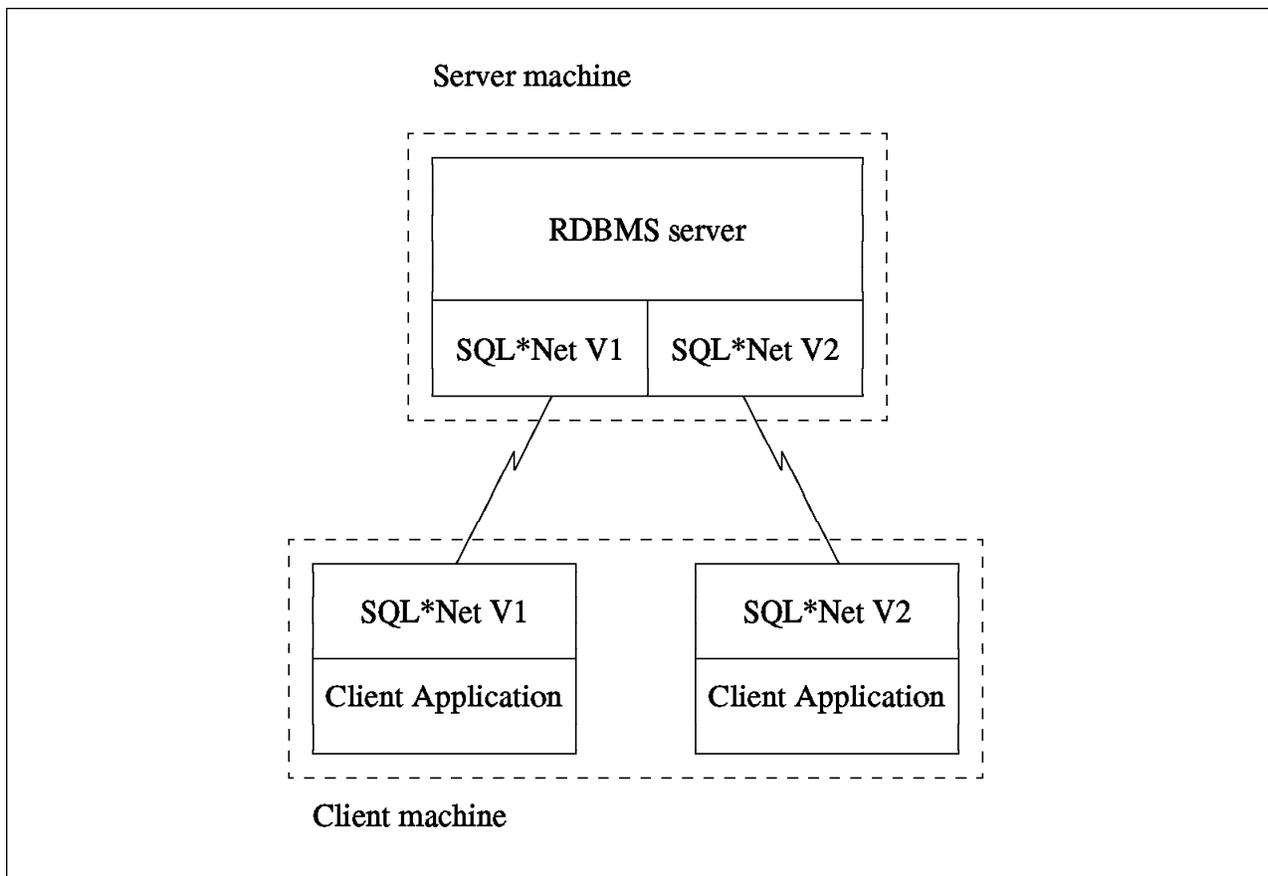
Figure 57. Client/Server Oracle Environment

5.8.2.1 Migrating the Database Server (Upgrade RDBMS to V7.1.6)

The procedure to upgrade the database server is the same as described in section 5.6, "Upgrading Oracle V6 and V7 to V7.1.6" on page 249.

5.8.2.2 Migrating Oracle on the Client

Since the client machine does not have the database, you do not have to perform the database migration. You just install the new release of Oracle tools if a new release is required to support the RDBMS server V7.1.6.

We recommend that you create a new ORACLE_HOME directory for the new release. If you choose to install the new release on the existing ORACLE_HOME directory, you should remove the old release first, and then install Oracle products with the following procedure:

- Change the environment variables:

  Make a decision on which directory to install the new release, and then change ORACLE_HOME and PATH environment variables to point to that directory.

- Install the new Oracle:

  Copy the Oracle Installer to a temporary directory and start the Installer. When the Installer prompts for ORACLE_HOME, enter the new ORACLE_HOME directory. Select **Install/Upgrade/Patch Software Only** in the Install Actions screen. Then, select all the Oracle products that you want to install.

- Modify your application:

  If your application refers to the ORACLE_HOME directory, change your source code to update the new ORACLE_HOME. You may have to change your application to support the new release and to exploit the new features. Please consult Oracle support for details.

5.8.2.3 Migrating SQL*Net (Upgrade SQL*Net to V2)

To upgrade SQL*Net to V2, you have to install SQL*Net V2 on both server and client machines. SQL*Net V2 provides a configuration tool that helps the database administrator to generate configuration files. Then you have to change the database connection string used in SQL*Net V1 to the new SQL*Net V2 format.

Please consult your Oracle support, and see the *SQL*Net Version 2.0 Administrator's Guide* for more details.

5.8.2.4 Migrating AIX on the Server

Use the same procedure as described in section 5.7, "Migrating an AIX V3.2.5 System with Oracle V7.1.6" on page 265 to upgrade AIX on the server machine to V4.1.3.

5.8.2.5 Migrating AIX on the Clients

The procedure to upgrade a client machine is slightly different from the procedure for the server machine. Since a client machine does not contain any database, you can eliminate the database backup step.

Use the following steps to upgrade the client machine

- Perform an AIX system backup:

  Use SMIT or the `mksysb` command to back up AIX files to a backup media. Make sure that you have mounted all file systems in rootvg. See InfoExplorer for more details.

- Migrate AIX to V4.1.3:

  Install AIX V4.1 using the Migration Install method. Refer to the *AIX V4.1 Installation Guide* and Chapter 1, "Migrating AIX from V3.2.5 to V4.1" on page 3 in this document for more details.

- Follow the instructions described in section 5.7.4, "Post Migration Steps" on page 268.

For recovery, follow these steps:

- Restore AIX:

  Reinstall AIX by booting the system with a system backup tape. When AIX shows the installation menu on the screen, select option 2, **Install a system that was created with the SMIT "Backup The System"**.

- Follow the instructions described in section 5.7.6.3, "Post-Restore steps" on page 273.

5.9 Exploiting AIX V4.1 Features

After the migration to AIX V4.1, you might want to exploit some of the features that are explained in the following sections.

5.9.1 LVM and File Systems

> ┌─ **Attention!** ──
>
> If you use the new features of the AIX V4.1 LVM, for example if you create a new logical volume with disk striping or a new compressed filesystem, the corresponding volume group will not be able to work with AIX V3.2. So, do not use this new LVM feature unless you are sure that the migration is completed, and you will not be returning to AIX V3.2.
>
> └───

1. Disk striping:

   If you use LVM disk striping, AIX will evenly distribute data blocks across disks to increase read/write throughput in a large I/O request. This feature is applicable for data files stored in both an AIX filesystem and a raw device (logical volume) since the disk striping parameter is specified at the logical volume level. It will help distribute the workload across several hard disks and should improve performance in I/O-intensive environments.

   There are some limitations in the current implementation:

   - A logical volume cannot be mirrored and striped at the same time. If a logical volume has mirroring and you want to distribute the workload across several disks, follow Oracle OFA rules to separate data, index, rollback segment, and redo log across different tablespaces.

   - The physical partitions for a striped logical volume must be evenly distributed among the disks. For example, if you have an existing 40 MB data file and want to distribute data on four disks, you have to create 3 physical partitions (PPs) on each disk. The total number of PPs will be 12, which represents 48 MB. Thus, 8 MB will be unused, unless you recreate Oracle data files to use 48 MB.

   - After a logical volume has been created, stripe size cannot be changed.

2. Very large file systems (>2 GB):

   This feature allows system administrators to create a filesystem larger than 2 GB. The maximum limit for the current implementation is 64 GB per filesystem. However, the file itself still has a maximum limit of 2 GB. You can utilize this feature by increasing the filesystem size. The Oracle applications can continue to work without any modification.

3. Compressed filesystems:

   The data compression will be done by AIX and then written to hard disk. The physical disk usage may be less than the logical files size. Users will gain benefit from saving disk space, but it will cost processor time. Also, the compression factor depends on the type of data in the file.

   In an environment with large text documents kept in a database file, this feature may save a lot of disk space. On the other hand this feature might not be suitable for a transaction environment where response time is the priority.

4. Fragmented blocks:

This feature allows you to change the block size from the 4096-bytes default to a smaller size that can be specified as 512, 1024 or 2048 bytes. File systems containing a large number of small files and directories will gain benefit from this space-saving feature.

In a database environment, the system usually contains only a small number of very large data files. Although you can keep a database in a filesystem with a small, fragmented size, the filesystem will not gain significant saving from it. However, Oracle product distribution, which is comprised of programs, libraries, script files, and so on, has some small files. In the testing environment that we used, we could save 1-2 percent of disk space by moving Oracle products to a new filesystem with a 512-byte fragmented size.

See the following section for instructions on how to move Oracle data files to a new logical volume or filesystem in order to utilize these features.

5.9.2 Moving Data Files to Utilize AIX V4.1 Features

Since AIX V4.1 LVM features must be specified when a logical volume or a filesystem is first created, you cannot set disk striping, change the fragment size or the compression algorithm for an existing logical volume or a filesystem. So, you have to move or copy existing data to a new raw device or to a new filesystem.

The database administrator can decide to move every data file in the database or only some highly active files to the new logical volume. The Oracle database allow the database administrator to rename a data file with some restrictions. So, if you have enough disk space, you can keep two copies of the data file on disk and avoid backing up data to tape.

5.9.2.1 Moving Several Database Files

If you plan to move several database files to the new filesystem or logical volume. The easiest way to do it is to shut down the whole database, and make a change with the following procedure:

- Shut down the database:

  Stop user activities and shut down the database as usual. As Oracle owner, issue the following command:

  ```
  $ export ORACLE_SID=<database_name>
  $ sqldba lmode=y
  SQLDBA> connect internal
  SQLDBA> shutdown normal
  ```

- Create new files and copy data:

  If there is a limited free space in the system, back up and remove your old data files, and eventually remove the corresponding filesystem or raw device. Create the new filesystem or logical volume with the required AIX V4.1 features. Then restore data back to the new files. If you have enough space, you can create the new files and copy data from the old files to the new files.

  You can use different path names for the new data files. Use the following commands to change file names in the Oracle database.

  ```
  sqldba lmode=y
  SQLDBA> connect internal
  SQLDBA> startup mount
  SQLDBA> alter database rename file '<old_name>' to '<new_name>';
  ```

Repeat the last command (alter database) for each data file.

- Restart the database:

  Use the following command to start the database, and test the new environment.

```
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> startup
```

5.9.2.2 Moving One Data File

To move only one data file to a new filesystem or to a new logical volume, you can either stop the whole database or stop only one tablespace and keep the database running. However, to stop the tablespace, it must be idle. This means that nobody must use data inside that tablespace during that time. One tablespace may comprise of one or more data files. If the tablespace is set offline, all data files in that tablespace will be unaccessible.

If you have enough free space, you may keep the old data file, and create a new data file with a new name but with the same size.

Use the following procedure to move a data file in the database:

- Locate the tablespace and the datafile by doing the following:

```
$ sqldba lmode=y
SQLDBA> connect internal
SQLDBA> select file_name,tablespace_name from dba_data_files;
```

  where file_name and tablespace_name are column names in the dba_data_files table.

- Set the tablespace offline to stop user activity to the data files;

```
SQLDBA> alter tablespace <tablespace_name> offline;
```

  where <tablespace_name> is the name of the tablespace.

- Create a new file, and make a copy of it:

  1. If you are going to use the same file name:

     – Copy data from the old file to the backup media.

     – Remove the old file.

     – Create a new logical volume or a new filesystem.

     – Restore data from the backup media to the new file.

  2. If you are going to use a different file name:

     – Create a new logical volume or filesystem.

     – Copy data from the old file to the new file.

     – Rename the data file in the Oracle database by using the following command:

```
SQLDBA> alter tablespace <tablespace_name> rename
        datafile '<old_file>' to '<new_file>';
```

- Set the tablespace online:

```
SQLDBA> alter tablespace <tablespace_name> online;
```

Note: For a new logical volume (or a raw device), you have to change its owner and the permission of the file in /dev. Change the owner to oracle, the group to dba and permissions to 600.

Appendix A. AIX V4.1 Packaging

The following sections describe the packaging of AIX V4.1.

A.1.1 Packaging Terminology

It is useful to understand the terminology as it relates to the packaging of AIX Version 4.1 and associated products.

Fileset is the POSIX term for the smallest installable unit within a product. A fileset is part of a package. For example, bos.net.tcp.client is a fileset for the bos.net package. This new packaging allows installation of only what is necessary (one or several specific filesets), generally requiring less disk space than AIX Version 3.

A **package** is a collection of filesets that are built together to form one installable image as a Backup File Format (BFF) file. For example, bos.net is a package.

Licensed Program Products (LPPs) are purchasable products. They can be a collection of packages or a single package. For example, items such as SNA, XLC are all LPPs. An LPP does not have to be contained in a single Backup File Format (BFF) image.

Fileset Update is an update that corrects or enhances function in a previously installed fileset. Since each fileset can be serviced separately, fixes for AIX V4.1 (fileset updates) are smaller and more localized. These are equivalent to a subsystem update on AIX V3.2.

Bundle is a file that contains a number of filesets for installation. They can be thought of as an installation profile. A number of bundles are supplied with AIX Version 4.1 for various environments, for example, client, server and application development. A system administrator can create his own bundles if the default bundles are not suitable.

Update Bundle is a collection of fixes and enhancements that updates the installed software to the latest level available on the media.

Maintenance Bundle is a predetermined level of fixes and enhancements for the Base Operating System (BOS). It is equivalent to the PMP (Preventive Maintenance Package) levels on AIX V3.2.

Product Offering is a selected set of packages (or LPPs) which are shipped together on the same physical media.

A.1.2 Bundles

The AIX V4.1 supplied bundle files are stored in the /usr/sys/inst.data/sys_bundles directory, and the files are:

```
# ls /usr/sys/inst.data/sys_bundles
ASCII.autoi          Client.def           Pers-Prod.bnd
App-Dev.bnd          GOS.autoi            Server.bnd
App-Dev.def          Graphics-Startup.bnd Server.def
BOS.autoi            Hdwr-Diag.bnd
Client.bnd           Hdwr-Diag.def
```

Additional bundles can be created by the system administrator using the Install and Update Software Manager (Virtual Storage Manager), and these files are created in the /usr/sys/inst.data/user_bundles directory. These files are ASCII files and can also be edited.

Client

This bundle includes a set of BOS packages deemed to provide the most common client functionality. This bundle has a base component (Client.bnd) and a graphical component.

Server

This bundle installs packages and options that provide a more robust, full-function server. It essentially installs commonly used AIX server functionality as well as enhanced Reliability, Availability, Serviceability (RAS) functionality. This bundle also has a base component (Server.bnd) and a graphical component. The extended RAS support and hardware diagnostics are included in this bundle.

Personal Productivity

This bundle, Pers-Prod.bnd, installs packages and options that provide an enhanced *Personal Productivity* environment for graphical desktop users. It essentially includes the same type of functionality as the client bundle with the addition of Common Desktop Environment and COSE applets. This environment is only available if the X11 runtime is installed.

Application Development

This bundle, App-Dev.bnd, is essentially the same as the client bundle with the addition of the filesets containing application development tools like compilers/linkers, libraries and debuggers.

DCE Client

This bundle contains the software required to be a client in a Distributed Computing Environment (DCE) network.

Hardware Diagnostics

This bundle, Hdwr-Diag.bnd, includes the bos.diag and devices.base packages plus the diagnostics filesets for the devices that are discovered during BOS installation. Once this bundle is installed, the diagnostics filesets for new devices will also be installed when using `cfgmgr -i`.

Media-Defined Bundle

This is a new feature included in AIX Version 4.1.4. It allows bundles to be defined by the installation media. This allows Independant Software Vendors (ISVs) to provide bundled installation of their software.

Other

Update, Maintenance Level and All can be used as *bundles* for installation.

All these bundles are available on the both types of packages - AIX Version 4.1 for Clients and AIX Version 4.1 for Servers. There will be a difference to what is installed as some of the filesets are not available on the Client media.

A.1.3 Fileset Names

With the repackaging of AIX V4.1, the operating system was divided into a lot of smaller components or filesets. All the of the filesets were renamed for AIX V4.1.

The following conventions were used when naming a software package and its filesets for AIX V4.1:

- All the fileset names must be of the form Package_Name.Option where Option is unique for the software product package, and Package_Name is the name of the software product package or LPP name. For example, *bos* or *X11*.

- The packages for a given product should begin with the product name followed by a dot (.).

- If a package has only one installable fileset, then the fileset name may be the same as the product name.

- All package names must be unique. Two software packages with the same name is not allowed.

- All fileset names must be made up of ASCII characters.

- Fileset names must be greater than one character in length and must begin with a letter or an underscore (_). Subsequent characters must be a letter, a digit, an underscore, or a dot (.).

- The maximum length for a fileset name is 144 bytes.

A.1.4 Standard Fileset Names

Standard fileset name extensions exist for certain types of filesets to help identify their usage. There is no requirement that any of these names be used in the name of a fileset.

.rte	Runtime or minimum set of commands and libraries
.adt	Application development portion of a product
.data	/usr/share portion of a product
.fnt	Font portion of a product
.diag	Diagnostics for a product
.ucode	Microcode for a product
.smit	SMIT tools and dialogues for a product
.mp	SMP unique code
.up	Uniprocessor unique code
.compat	Compatibility code (to be removed in a future release)
.loc	Locale files for a specific fileset
.msg.(lang)	Message catalogues for a specific fileset

.info.(lang)   InfoExplorer databases for a specific fileset
.help.(lang)   Help dialogs for a specific fileset

A.1.5 Device Driver Packaging

The AIX V4.1 Configuration Manager (cfgmgr) will automatically install software support for detectable devices. The naming convention for device driver packaging is *devices.bus_type.card_id* and where the *bus_type* is:

mca   for MicroChannel
pci   for PCI Bus
isa   for ISA Bus
sys   for RSC Bus
buc   for 601 Bus
sio   for system planar
pcmcia for PCMCIA devices

card_id is the unique hexadecimal card identifier.

For example, the package name for the microchannel FDDI adapter device support is devices.mca.8ef4, and the package name for parallel printer port support is devices.sio.ppa. A list of device drivers filesets with card identifier is provided in Appendix G, "AIX V4.1 Device Driver Packages" on page 319.

Each adapter device driver is pakaged in three parts: the config method and device driver fileset, the diagnostics fileset and the microcode fileset. For example, the filesets for the FDDI adapter will be called:

devices.mca.8ef4.rte   Device driver code and config methods
devices.mca.8ef4.diag   Diagnostic code for FDDI
devices.mca.8ef4.ucode   Microcode for FDDI

sys is the RISC Single Chip (RSC) bus that is used to connect the graphic adapters in the RISC System/6000 Models 220 and 230. **buc** is the PowerPC 601 bus that is used to connect the graphic adapters in the RISC System/6000 Models 25x and 41x. **sio** is used for the devices that are connected to the system planar, such as diskette drives, serial and parallel ports, keyboard, and mouse.

Note: Install all the device drivers or a superset of the device support required for all your systems if you are cloning AIX V4.1 systems with mksysb.

A.1.6 Message Catalog Packaging

The messages, locales, convertors, help, and info databases are all shipped by language. The language in a fileset name is commonly indicated by the *lang* wildcard since most language-dependent filesets are translated into more than one language.

The naming convention for all message packages shipped in AIX V4.1 is:
LPP.msg.(LANG)(.Fileset_Name)

where *LANG* is any of the supported languages. Fileset_Name is optional since there may be only one message fileset for the LPP.

Some examples :

- bos.net.nfs.client messages will be packaged in bos.msg.(lang).net.nfs.client.

- bos.msg.en_US.net.tcp.client is TCP/IP Client Support Messages for US English.

AIX V4.1 will automatically install the message filesets for the primary language for filesets that use this naming convention. The primary language can be specified during BOS installation. Additional language filesets can be installed after the system has been installed.

This naming convention is also used for the other language-dependent filesets, such as InfoExplorer databases, COSE help files and language converters.

A.1.7 Package Installation Database

A fileset called *pkg_gd* (Package Guide) is shipped on the installation media to provide the current information on the various products available for AIX V4.1. It is not installed automatically; so you have to manually install it from the installation media. It contains information about:

- Package and Fileset Names for LPPs available on AIX V4.1

- Approximate Disk Space Required

- Requisite Software

- Special Installation Notes

- Special Migration Notes

To view the database, you need to issue the command `info -l lp_info`.

Note: The pkg_gd fileset is shipped with AIX V4.1.3, but is not shipped with AIX V4.1.4. It will be shipped in future releases of AIX.

Appendix B. AIX V4.1 Software Maintenance

In AIX V4.1, the software maintenance levels are identified by Version Release Modification Fix (VRMF) levels. The VRMF levels are displayed by using the `lslpp` command. You will no longer see any (PTF) Program Temporary Fix (U4xxxxx) numbers. For example, in AIX V4.1.4, bos.rte.up fileset has a VRMF of 4.1.4.0, indicating Version 4, Release 1, Modification level 4, and Fix level 0.

The **Version** number is incremented to indicate a new product or the repackaging of an existing product. Versions include major functional enhancements and typically come two or more years apart.

The **Release** number is incremented to indicate new enhancements or new functions. Releases of AIX V4.1 comes approximately one year apart.

The **Modification** number is incremented whenever an accumulation of maintenance is added to a fileset. A modification level can also include support for new processors or devices where this support does not affect the behavior of the product on existing systems. Whenever the modification level is adjusted, the fix level is reset to zero. The modification levels for AIX V4.1 comes 3-6 months apart.

The **Fix** level is incremented whenever a fix is added to the fileset. Fixes for AIX V4.1 are created on customer demand.

Each fileset in AIX V4.1 can be serviced separately. Fixes will be delivered in fileset packages. Changes to filesets are cumulative, meaning that each new level of a fileset contains all the previous changes.

Maintenance and fixes should not change application programming interfaces so that applications that are written to the documented programming interfaces will function identically on different maintenance and fix levels. This is important for software vendors or customers for certification of their applications on AIX V4.1.

B.1.1 Fileset VRMF Numbering

The result of the `lslpp -L` command will show that the various filesets have different VRMF for a particular level of AIX V4.1. In AIX V4.1.4, not all of the filesets are at level 4.1.4.0.

```
# lslpp -L
  Fileset                   Level  State  Description
  ----------------------------------------------------------------------------
  bos.rte.up                4.1.4.0  C    Base Operating System
                                          Uniprocessor Runtime
  bos.sysmgt.serv_aid       4.1.4.0  C    Software Error Logging and Dump
                                          Service Aids
  bos.sysmgt.smit           4.1.4.0  C    System Management Interface Tool
                                          (SMIT)
  bos.sysmgt.sysbr          4.1.4.0  C    System Backup and BOS Install
                                          Utilities
  bos.terminfo.com.data     4.1.0.0  C    Common Terminal Definitions
  bos.terminfo.dec.data     4.1.3.0  C    Digital Equipment Corp. Terminal
                                          Definitions
  bos.terminfo.ibm.data     4.1.4.0  C    IBM Terminal Definitions
```

```
bos.terminfo.pc.data      4.1.0.0   C    Personal Computer Terminal
                                          Definitions
bos.terminfo.print.data   4.1.0.0   C    Generic Line Printer Terminal
                                          Definitions
bos.terminfo.rte          4.1.3.0   C    Run-time Environment for AIX
                                          Terminals
connect.client            1.1.0.0   C    Connections Client Software
connect.msg.en_US         1.1.0.0   C    Connections Server and Client
                                          Messages - U. S. English
connect.protocols         1.1.0.0   C    NW, LS, MAC, and Client
                                          Protocols
connect.server.com        1.1.0.0   C    Connections Common Server Files
connect.server.lsserve    1.1.0.0   C    LS_Server Software
connect.server.macserve   1.1.0.0   C    MAC_Server Software
connect.server.nwserve    1.1.0.0   C    NW_Server Software
```

If the fileset for the base operating system was shipped in 4.1.0 and if there were absolutely no code changes to that fileset since 4.1.0, the VRMF is unchanged. For example, bos.terminfo.com.data fileset is still at the VRMF level 4.1.0.0 in AIX V4.1.4.

If the fileset for the base operating system was shipped in 4.1.0 and if there were changes to the code for 4.1.1 or later, the modification level for the VRMF will be adjusted to be inline with the operating-system level. For example, the level of bos.rte.up was 4.1.1.0 for AIX V4.1.0 and 4.1.2.0 for AIX V4.1.2 and is 4.1.4.0 for AIX V4.1.4.

If there is a new (additional) fileset for the base operating system or an LPP, the VRMF will be the level of the operating system or the LPP that it ships with.

If the LPP is new for V4.1, the VRMF level can really be set to any version or release value (but probably 1.1.0.0).

Figure 58 on page 291 is an illustration of the fileset numbering mechanism.
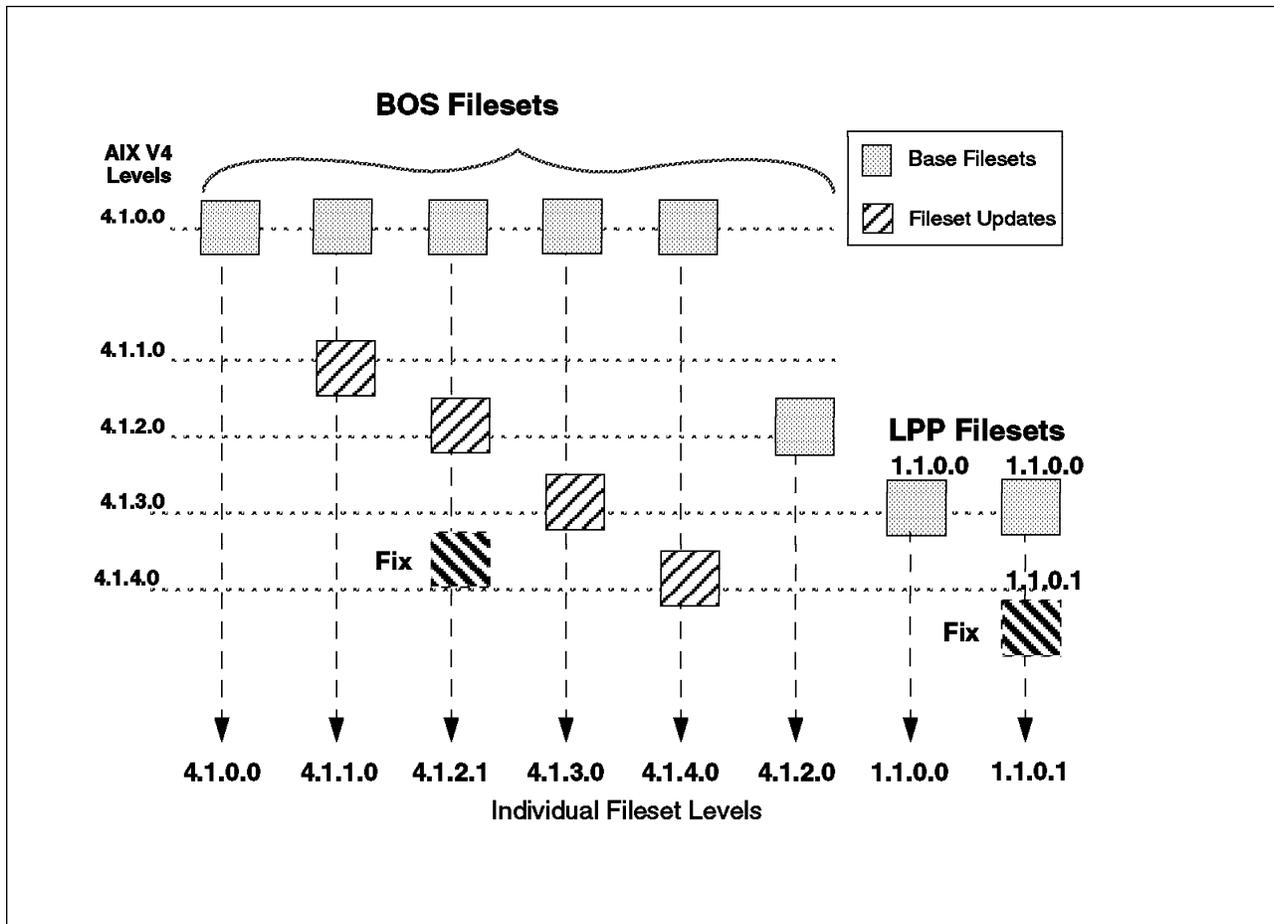
Figure 58. Fileset Numbering Examples

B.1.2 oslevel Command

The `oslevel` command has been retained from AIX V3.2.4. Its behavior is slightly different due to the new VRMF format (and it is quicker!).

```
# oslevel -?
Usage: oslevel ( -l <level> | -g | -q )
    -l : List filesets at levels earlier than maintenance level
         specified by the <level> parameter
    -g : List filesets at levels later than most recent
         complete maintenance level
    -q : List names of known maintenance levels which may be
         specified with the -l flag

  Output indicates that base system software is entirely at
  or above a particular maintenance level. Corresponding output
  would be 4.1.1.0 first AIX V4.1 maintenance level.

  The additional options may be specified to determine which
  filesets differ from the maintenance level
```

To check the level of the operating system, enter:

```
# oslevel
4.1.4.0
```

To list the names of the known Maintenance Levels, enter:

```
# oslevel -q
Known Maintenance Levels
------------------------
4.1.4.0
#
```

The VRMF information that the `oslevel` command uses for all the base operating
system filesets is now stored in an ODM database. There is a specific entry in the
`/usr/lib/objrepos/fix` database for each `Maintenance Level`. It looks like this:

```
# ODMDIR=/usr/lib/objrepos odmget fix | pg

fix:
    name = "4.1.4.0_AIX_ML"
    abstract = "AIX 4.1.4.0 Maintenance Level"
    type = "p"
    filesets = "X11.Dt.ToolTalk:4.1.4.0\n\
X11.Dt.adt:4.1.4.0\n\
X11.Dt.bitmaps:4.1.4.0\n\
X11.Dt.compat:4.1.4.0\n\
X11.Dt.helpinfo:4.1.3.0\n\
X11.Dt.helpmin:4.1.3.0\n\
X11.Dt.helprun:4.1.3.0\n\
X11.Dt.lib:4.1.4.0\n\
X11.Dt.rte:4.1.4.0\n\
X11.Dt.xdt2cde:4.1.3.0\n\
X11.adt.bitmaps:4.1.0.0\n\
X11.adt.ext:4.1.4.0\n\
X11.adt.imake:4.1.3.0\n\
X11.adt.include:4.1.3.0\n\
X11.adt.lib:4.1.1.0\n\
X11.adt.motif:4.1.4.0\n\
X11.apps.aixterm:4.1.3.0\n\
X11.apps.clients:4.1.3.0\n\
 ...
```

B.1.3 instfix Command

There is a new `instfix` command with AIX V4.1, and there are a number of
options to this command. It can be used to:

- List the contents of an update media

- Search for a fix number on a media

- Search in ODM for installed fixes

- Search for a key word

- Give fix abstracts

Following are some examples:

To list the entire fixes Table of Contents on the media, enter:

```
# instfix -T -d/dev/rmt0.1
```

To install all filesets associated with fix IX38794 from the tape mounted on /dev/rmt0.1, enter:

```
# instfix -k IX38794 -d /dev/rmt0.1
```

To install all fixes on the media in the tape drive, enter:

```
# instfix -T -d /dev/rmt0.1 | instfix -d /dev/rmt0.1 -f
```

The first part of this command lists the fixes on the media, and the second part of this command uses the list as input.

To list all entries on the tape using a keyword search string of SCSI, enter:

```
# instfix -s SCSI -d /dev/rmt0.1
```

To inform the user on whether fixes IX38794 and IX48523 are installed, enter:

```
# instfix -i -k "IX38794 IX48523"
```

B.1.4 Applying Updates

There are a number of SMIT fastpaths to enable you to install maintenance to your AIX V4.1 system.

`smitty update_all` will install all the fixes that are available on the installation media.

`smitty install_fileset` allows the system administrator to install individual fixes. Use / to search for the fix and PF7 to select multiple filesets.

Appendix C. AIX V4.1 Compatibility Filesets

You will find below the content of the main compatibility filesets for AIX V4.1.4.

C.1 bos.compat.cmds Fileset

```
    Fileset   File
    --------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.cmds 4.1.3.0
              /usr/lib/security/scan/xa1clean.lst
              /usr/lib/security/scan/virsig.lst
              /usr/bin/msgs
              /usr/bin/del
              /usr/bin/page -> /usr/bin/more
              /usr/bin/copy -> /usr/bin/cp
              /usr/bin/virscan
              /usr/lib/security/scan
              /usr/lib/security/scan/ghoclean.lst
              /usr/bin/move -> /usr/bin/mv
              /usr/bin/joinconf -> /usr/bin/confer
              /usr/bin/confer
              /usr/lib/security/scan/648clean.lst
              /usr/lib/security/scan/darclean.lst
              /usr/ucb/more -> /usr/bin/more
              /usr/lib/security/scan/vacclean.lst
              /usr/bin/di -> /usr/bin/li
              /usr/bin/li
              /usr/bin/delete -> /usr/bin/rm
              /usr/ucb/sccs -> /usr/bin/sccs
```

C.2 bos.compat.lan Fileset

```
    Fileset   File
    --------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.lan 4.1.3.0
              /usr/lib/methods/ciedd.cat
              /usr/include/sys/fddiuser.h
              /usr/lib/methods/ucfgcie
              /usr/lib/methods/cfgcie
              /usr/lib/drivers/ciedd
```

C.3 bos.compat.libs Fileset

```
    Fileset   File
    --------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.libs 4.1.1.0
              /usr/lib/lib300.a
              /usr/lib/lib300S.a -> /usr/lib/lib300s.a
              /usr/lib/lib300s.a
              /usr/lib/libplot.a
              /usr/lib/lib4014.a
```

```
                   /usr/lib/lib450.a
                   /usr/lib/libprint.a
```

C.4 bos.compat.links Fileset

```
        Fileset    File
        ----------------------------------------------------------------------------
   Path: /usr/lib/objrepos
     bos.compat.links 4.1.0.0
                   /usr/ucb/fmt -> /usr/bin/fmt
                   /usr/bin/lsnamsv -> /usr/sbin/lsnamsv
                   /usr/bin/chroot -> /usr/sbin/chroot
                   /usr/ucb/fold -> /usr/bin/fold
                   /usr/bin/chrctcp -> /usr/sbin/chrctcp
                   /usr/bin/mktcpip -> /usr/sbin/mktcpip
                   /usr/etc
                   /usr/ucb/lastcomm -> /usr/bin/lastcomm
                   /usr/ucb/vmstat -> /usr/bin/vmstat
                   /usr/ucb/ex -> /usr/bin/vi
                   /usr/ucb/man -> /usr/bin/man
                   /usr/ucb/sysline -> /usr/bin/sysline
                   /usr/ucb/biff -> /usr/bin/biff
                   /usr/ucb/ftp -> /usr/bin/ftp
                   /usr/ucb/rlogin -> /usr/bin/rlogin
                   /usr/ucb/uptime -> /usr/bin/w
                   /usr/bin/rmgroup -> /usr/sbin/rmgroup
                   /usr/bin/sag -> /usr/sbin/sag
                   /usr/ucb/tftp -> /usr/bin/tftp
                   /usr/bin/sar -> /usr/sbin/sar
                   /usr/bin/sync -> /usr/sbin/sync
                   /usr/ucb/ctags -> /usr/bin/ctags
                   /usr/usg/install -> /usr/bin/install
                   /usr/ucb/whoami -> /usr/bin/whoami
                   /usr/ucb/quota -> /usr/sbin/quota
                   /usr/ucb/script -> /usr/bin/script
                   /usr/bin/ipreport -> /usr/sbin/ipreport
                   /usr/ucb/head -> /usr/bin/head
                   /usr/ucb/leave -> /usr/bin/leave
                   /usr/bin/chprtsv -> /usr/sbin/chprtsv
                   /usr/ucb/logger -> /usr/bin/logger
                   /usr/usg/tr -> /usr/bin/tr
                   /usr/ucb/tn -> /usr/bin/telnet
                   /usr/ucb/view -> /usr/bin/vi
                   /usr/ucb/netstat -> /usr/sbin/netstat
                   /usr/ucb/quotaoff -> /usr/sbin/quotaon
                   /usr/ucb/ul -> /usr/bin/ul
                   /usr/ucb/vi -> /usr/bin/vi
                   /usr/ucb/compress -> /usr/bin/compress
                   /usr/bin/rmnamsv -> /usr/sbin/rmnamsv
                   /usr/bin/lsuser -> /usr/sbin/lsuser
                   /usr/ucb/edit -> /usr/bin/vi
                   /usr/ucb/wc -> /usr/bin/wc
                   /usr/bin/hostid -> /usr/sbin/hostid
                   /usr/etc/rpc.lockd -> /usr/sbin/rpc.lockd
                   /usr/bin/format -> /usr/sbin/format
                   /usr/ucb/whois -> /usr/bin/whois
                   /usr/bin/mkprtsv -> /usr/sbin/mkprtsv
```

```
/usr/bin/grpck -> /usr/sbin/grpck
/usr/ucb/vacation -> /usr/bin/vacation
/usr/bin/strace -> /usr/sbin/strace
/usr/ucb/msgs -> /usr/bin/msgs
/usr/ucb/tcopy -> /usr/bin/tcopy
/usr/ucb/whereis -> /usr/bin/whereis
/usr/bin/bosboot -> /usr/sbin/bosboot
/usr/ucb/groups -> /usr/bin/groups
/usr/ucb/reset -> /usr/bin/tset
/usr/ucb/zcat -> /usr/bin/compress
/usr/ucb/uncompress -> /usr/bin/compress
/usr/ucb/which -> /usr/bin/which
/usr/ucb/rdist -> /usr/bin/rdist
/usr/ucb/from -> /usr/bin/from
/usr/bin/rmuser -> /usr/sbin/rmuser
/usr/bin/strerr -> /usr/sbin/strerr
/usr/ucb/ruptime -> /usr/bin/ruptime
/usr/ucb/page -> /usr/bin/more
/usr/ucb/quot -> /usr/sbin/quot
/usr/ucb/apply -> /usr/bin/apply
/usr/ucb/expand -> /usr/bin/expand
/usr/ucb/f -> /usr/bin/finger
/usr/ucb/u -> /usr/bin/users
/usr/ucb/w -> /usr/bin/w
/usr/ucb/tn3270 -> /usr/bin/telnet
/usr/ucb/tail -> /usr/bin/tail
/usr/etc/portmap -> /usr/sbin/portmap
/usr/ucb/finger -> /usr/bin/finger
/usr/bin/chnamsv -> /usr/sbin/chnamsv
/usr/ucb/talk -> /usr/bin/talk
/usr/ucb/printenv -> /usr/bin/env
/usr/bin/ff -> /usr/sbin/ff
/usr/ucb/mailq -> /usr/sbin/sendmail
/usr/bin/lsprtsv -> /usr/sbin/lsprtsv
/usr/ucb/last -> /usr/bin/last
/usr/ucb/Mail -> /usr/bin/mail
/usr/ucb/edquota -> /usr/sbin/edquota
/usr/ucb/rsh -> /usr/bin/rsh
/usr/bin/pac -> /usr/sbin/pac
/usr/bin/fdformat -> /usr/sbin/format
/usr/ucb/syscall -> /usr/bin/syscall
/usr/ucb/apropos -> /usr/bin/man
/usr/bin/mknamsv -> /usr/sbin/mknamsv
/usr/ucb/newaliases -> /usr/sbin/sendmail
/usr/ucb/yes -> /usr/bin/yes
/usr/lib/mailstats -> /usr/sbin/mailstats
/usr/bin/lsgroup -> /usr/sbin/lsgroup
/usr/bin/iptrace -> /usr/sbin/iptrace
/usr/ucb/users -> /usr/bin/users
/usr/ucb/utftp -> /usr/bin/tftp
/usr/ucb/strings -> /usr/ccs/bin/strings
/usr/ucb/repquota -> /usr/sbin/repquota
/usr/ucb/whatis -> /usr/bin/man
/usr/ucb/clear -> /usr/bin/clear
/usr/ucb/grep -> /usr/bin/grep
/usr/ucb/rwho -> /usr/bin/rwho
/usr/ucb/remsh -> /usr/bin/rsh
/usr/bin/wall -> /usr/sbin/wall
```

```
                    /usr/ucb/colrm -> /usr/bin/colrm
                    /usr/bin/killall -> /usr/sbin/killall
                    /usr/ucb/unexpand -> /usr/bin/unexpand
                    /usr/bin/strclean -> /usr/sbin/strclean
                    /usr/lib/getNAME -> /usr/lbin/getNAME
                    /usr/bin/lsadpnm -> /usr/sbin/lsadpnm
                    /usr/lib/diff3prog -> /usr/lbin/diff3prog
                    /usr/ucb/tset -> /usr/bin/tset
                    /usr/ucb/quotacheck -> /usr/sbin/quotacheck
                    /usr/bin/lptest -> /usr/sbin/lptest
                    /usr/ucb/quotaon -> /usr/sbin/quotaon
                    /usr/ucb/lock -> /usr/bin/lock
                    /usr/ucb/telnet -> /usr/bin/telnet
                    /usr/bin/rmprtsv -> /usr/sbin/rmprtsv
                    /usr/ucb/sendbug -> /usr/bin/sendbug
                    /usr/etc/rpc.statd -> /usr/sbin/rpc.statd

          Path: /etc/objrepos
            bos.compat.links 4.1.0.0
                    /etc/srcmstr -> /usr/sbin/srcmstr
                    /etc/savecore -> /usr/sbin/savecore
                    /etc/mkvirprt -> /usr/sbin/mkvirprt
                    /etc/lvrelminor -> /usr/sbin/lvrelminor
                    /etc/wall -> /usr/sbin/wall
                    /etc/putlvcb -> /usr/sbin/putlvcb
                    /etc/talkd -> /usr/sbin/talkd
                    /etc/bffcreate -> /usr/sbin/bffcreate
                    /etc/accton -> /usr/sbin/acct/accton
                    /etc/htable -> /usr/sbin/htable
                    /etc/getty -> /usr/sbin/tsm
                    /etc/crvfs -> /usr/sbin/lsvfs
                    /etc/objrepos/sm_name_hdr.vc -> /usr/lib/objrepos/sm_name_hdr.vc
                    /etc/rlogind -> /usr/sbin/rlogind
                    /etc/swcons -> /usr/sbin/swcons
                    /etc/execerror -> /usr/sbin/execerror
                    /etc/lvrelmajor -> /usr/sbin/lvrelmajor
                    /etc/lreducelv -> /usr/sbin/lreducelv
                    /etc/varyoffvg -> /usr/sbin/varyoffvg
                    /etc/chlv -> /usr/sbin/chlv
                    /etc/mklost+found -> /usr/sbin/mklost+found
                    /etc/syslogd -> /usr/sbin/syslogd
                    /etc/rmlv -> /usr/sbin/rmlv
                    /etc/mkpasswd -> /usr/sbin/mkpasswd
                    /etc/rmvfs -> /usr/sbin/lsvfs
                    /etc/telnetd -> /usr/sbin/telnetd
                    /etc/no -> /usr/sbin/no
                    /etc/restbyinode -> /usr/sbin/restbyinode
                    /etc/mvdir -> /usr/sbin/mvdir
                    /etc/lsvfs -> /usr/sbin/lsvfs
                    /etc/rdump -> /usr/sbin/rdump
                    /etc/phones-file -> /usr/lib/phones-file
                    /etc/cfgvg -> /usr/sbin/cfgvg
                    /etc/chps -> /usr/sbin/lsps
                    /etc/chpv -> /usr/sbin/chpv
                    /etc/chitab -> /usr/sbin/mkitab
                    /etc/rmps -> /usr/sbin/lsps
                    /etc/lqueryvgs -> /usr/sbin/lqueryvgs
                    /etc/reducevg -> /usr/sbin/reducevg
```

```
/etc/varyonvg -> /usr/sbin/varyonvg
/etc/sa -> /usr/sbin/sa
/etc/watch -> /usr/sbin/watch
/etc/link -> /usr/sbin/link
/etc/piopredef -> /usr/sbin/piopredef
/etc/getvgname -> /usr/sbin/getvgname
/etc/lcreatelv -> /usr/sbin/lcreatelv
/etc/lvaryonvg -> /usr/sbin/lvaryonvg
/etc/auditbin -> /usr/sbin/auditbin
/etc/lscfg -> /usr/sbin/lscfg
/etc/updatelv -> /usr/sbin/updatelv
/etc/chvg -> /usr/sbin/chvg
/etc/inuumsg -> /usr/sbin/inuumsg
/etc/lvmmsg -> /usr/sbin/lvmmsg
/etc/inudocm -> /usr/sbin/inudocm
/etc/splitlvcopy -> /usr/sbin/splitlvcopy
/etc/uprintfd -> /usr/sbin/uprintfd
/etc/mknod -> /usr/sbin/mknod
/etc/security/mkuser.sys -> /usr/lib/security/mkuser.sys
/etc/backup -> /usr/sbin/backup
/etc/cfgmgr -> /usr/sbin/cfgmgr
/etc/yp -> /var/yp
/etc/syncd -> /usr/sbin/syncd
/etc/getlvodm -> /usr/sbin/getlvodm
/etc/synclvodm -> /usr/sbin/synclvodm
/etc/ftpd -> /usr/sbin/ftpd
/etc/bootpd -> /usr/sbin/bootpd
/etc/fingerd -> /usr/sbin/fingerd
/etc/gettable -> /usr/sbin/gettable
/etc/remote-file -> /usr/lib/remote-file
/etc/inetd -> /usr/sbin/inetd
/etc/dfsck -> /usr/sbin/dfsck
/etc/fastboot -> /usr/sbin/fastboot
/etc/lchangelv -> /usr/sbin/lchangelv
/etc/lsvgfs -> /usr/sbin/lsvgfs
/etc/install -> /usr/bin/install
/etc/lsparent -> /usr/sbin/lsparent
/etc/lmigratepp -> /usr/sbin/lmigratepp
/etc/named-xfer -> /usr/sbin/named-xfer
/etc/lcreatevg -> /usr/sbin/lcreatevg
/etc/objrepos/sm_cmd_opt -> /usr/lib/objrepos/sm_cmd_opt
/etc/unlink -> /usr/sbin/link
/etc/dumpfs -> /usr/sbin/dumpfs
/etc/halt -> /usr/sbin/halt
/etc/rrestore -> /usr/sbin/rrestore
/etc/devnm -> /usr/sbin/devnm
/etc/updatevg -> /usr/sbin/updatevg
/etc/named -> /usr/sbin/named
/etc/objrepos/sm_cmd_hdr.vc -> /usr/lib/objrepos/sm_cmd_hdr.vc
/etc/lchangepv -> /usr/sbin/lchangepv
/etc/restbase -> /usr/lib/boot/restbase
/etc/pdisable -> /usr/sbin/penable
/etc/imfs -> /usr/sbin/lsfs
/etc/tftpd -> /usr/sbin/tftpd
/etc/reboot -> /usr/sbin/reboot
/etc/crash -> /usr/sbin/crash
/etc/importvg -> /usr/sbin/importvg
/etc/ndb -> /usr/sbin/ndb
```

```
/etc/skulker -> /usr/sbin/skulker
/etc/pdelay -> /usr/sbin/penable
/etc/rmt -> /usr/sbin/rmt
/etc/flcopy -> /usr/sbin/flcopy
/etc/mkdev -> /usr/sbin/mkdev
/etc/lsvirprt -> /usr/sbin/lsvirprt
/etc/mklvcopy -> /usr/sbin/extendlv
/etc/migratepv -> /usr/sbin/migratepv
/etc/getlvname -> /usr/sbin/getlvname
/etc/rmvirprt -> /usr/sbin/rmvirprt
/etc/tsm -> /usr/sbin/tsm
/etc/XNSrouted -> /usr/sbin/XNSrouted
/etc/lvaryoffvg -> /usr/sbin/lvaryoffvg
/etc/pstat -> /usr/sbin/crash
/etc/lsconn -> /usr/sbin/lsconn
/etc/copyrawlv -> /usr/sbin/copyrawlv
/etc/lscons -> /usr/sbin/lscons
/etc/getlvcb -> /usr/sbin/getlvcb
/etc/timed -> /usr/sbin/timed
/etc/mkhosts -> /usr/sbin/mkhosts
/etc/chdev -> /usr/sbin/chdev
/etc/mkfs -> /usr/sbin/mkfs
/etc/trpt -> /usr/sbin/trpt
/etc/linstallpv -> /usr/sbin/linstallpv
/etc/swapon -> /usr/sbin/swapon
/etc/XNSquery -> /usr/sbin/XNSquery
/etc/audit -> /usr/sbin/audit
/etc/cfgodm.ipl -> /usr/lib/cfgodm.ipl
/etc/nls -> /usr/lib/nls
/etc/renice -> /usr/sbin/renice
/etc/arp -> /usr/sbin/arp
/etc/security/mkuser.default -> /usr/lib/security/mkuser.default
/etc/asw -> /usr/lib/asw
/etc/redefinevg -> /usr/sbin/redefinevg
/etc/auditcat -> /usr/sbin/auditcat
/etc/mount -> /usr/sbin/mount
/etc/lresynclp -> /usr/sbin/lresynclp
/etc/lvgenminor -> /usr/sbin/lvgenminor
/etc/lresynclv -> /usr/sbin/lresynclv
/etc/ckprereq -> /usr/sbin/ckprereq
/etc/telinit -> /usr/sbin/init
/etc/mklv -> /usr/sbin/mklv
/etc/auditstream -> /usr/sbin/auditstream
/etc/exportvg -> /usr/sbin/exportvg
/etc/timedc -> /usr/sbin/timedc
/etc/chvirprt -> /usr/sbin/chvirprt
/etc/comsat -> /usr/sbin/comsat
/etc/tapechk -> /usr/sbin/tapechk
/etc/catman -> /usr/sbin/catman
/etc/putlvodm -> /usr/sbin/putlvodm
/etc/penable -> /usr/sbin/penable
/etc/fsck -> /usr/sbin/fsck
/etc/lvgenmajor -> /usr/sbin/lvgenmajor
/etc/inusave -> /usr/sbin/inusave
/etc/fsdb -> /usr/sbin/fsdb
/etc/reorgvg -> /usr/sbin/reorgvg
/etc/lresyncpv -> /usr/sbin/lresyncpv
/etc/lquerylv -> /usr/sbin/lquerylv
```

```
/etc/extendlv -> /usr/sbin/extendlv
/etc/rmdev -> /usr/sbin/rmdev
/etc/pshare -> /usr/sbin/penable
/etc/mkps -> /usr/sbin/lsps
/etc/umount -> /usr/sbin/mount
/etc/phold -> /usr/sbin/penable
/etc/objrepos/sm_menu_opt.vc -> /usr/lib/objrepos/sm_menu_opt.vc
/etc/allocp -> /usr/sbin/allocp
/etc/unmount -> /usr/sbin/mount
/etc/lsitab -> /usr/sbin/mkitab
/etc/crfs -> /usr/sbin/lsfs
/etc/shutdown -> /usr/sbin/shutdown
/etc/lvchkmajor -> /usr/sbin/lvchkmajor
/etc/lsdev -> /usr/sbin/lsdev
/etc/slibclean -> /usr/sbin/slibclean
/etc/fuser -> /usr/sbin/fuser
/etc/lextendlv -> /usr/sbin/lextendlv
/etc/objrepos/sm_cmd_opt.vc -> /usr/lib/objrepos/sm_cmd_opt.vc
/etc/routed -> /usr/sbin/routed
/etc/restore -> /usr/sbin/restore
/etc/lquerypv -> /usr/sbin/lquerypv
/etc/mergedev -> /usr/lib/boot/mergedev
/etc/inurecv -> /usr/sbin/inurecv
/etc/backbyinode -> /usr/sbin/backbyinode
/etc/lsfs -> /usr/sbin/lsfs
/etc/mkvg -> /usr/sbin/mkvg
/etc/restbyname -> /usr/sbin/restbyname
/etc/ip_auditpr -> /usr/sbin/ip_auditpr
/etc/gated -> /usr/sbin/gated
/etc/rwhod -> /usr/sbin/rwhod
/etc/fasthalt -> /usr/sbin/fasthalt
/etc/auditpr -> /usr/sbin/auditpr
/etc/rexecd -> /usr/sbin/rexecd
/etc/pstart -> /usr/sbin/penable
/etc/lqueryvg -> /usr/sbin/lqueryvg
/etc/extendvg -> /usr/sbin/extendvg
/etc/ncheck -> /usr/sbin/ncheck
/etc/cron -> /usr/sbin/cron
/etc/lvlstmajor -> /usr/sbin/lvlstmajor
/etc/rmitab -> /usr/sbin/mkitab
/etc/rshd -> /usr/sbin/rshd
/etc/cplv -> /usr/sbin/cplv
/etc/tstresp -> /usr/sbin/tstresp
/etc/lslv -> /usr/sbin/lslv
/etc/objrepos/sm_menu_opt -> /usr/lib/objrepos/sm_menu_opt
/etc/ldeletelv -> /usr/sbin/ldeletelv
/etc/syncvg -> /usr/sbin/syncvg
/etc/ipl_varyon -> /usr/sbin/ipl_varyon
/etc/security/scan -> /usr/lib/security/scan
/etc/migfix -> /usr/sbin/migfix
/etc/slattach -> /usr/sbin/slattach
/etc/mkproto -> /usr/sbin/mkproto
/etc/lsattr -> /usr/sbin/lsattr
/etc/objrepos/sm_cmd_hdr -> /usr/lib/objrepos/sm_cmd_hdr
/etc/diag -> /usr/sbin/diag
/etc/lsps -> /usr/sbin/lsps
/etc/rmlvcopy -> /usr/sbin/rmlvcopy
/etc/lspv -> /usr/sbin/lspv
```

```
                       /etc/ac -> /usr/sbin/acct/ac
                       /etc/ldeletepv -> /usr/sbin/ldeletepv
                       /etc/proto -> /usr/sbin/proto
                       /etc/qdaemon -> /usr/sbin/qdaemon
                       /etc/chown -> /usr/bin/chown
                       /etc/backbyname -> /usr/sbin/backbyname
                       /etc/chvfs -> /usr/sbin/lsvfs
                       /etc/chcons -> /usr/sbin/chcons
                       /etc/chtcb -> /usr/sbin/chtcb
                       /etc/inurest -> /usr/sbin/inurest
                       /etc/savebase -> /usr/sbin/savebase
                       /etc/auditselect -> /usr/sbin/auditselect
                       /etc/lsvg -> /usr/sbin/lsvg
                       /etc/lmktemp -> /usr/sbin/lmktemp
                       /etc/mkitab -> /usr/sbin/mkitab
                       /etc/chfs -> /usr/sbin/lsfs
                       /etc/rmfs -> /usr/sbin/lsfs
                       /etc/objrepos/sm_name_hdr -> /usr/lib/objrepos/sm_name_hdr
                       /etc/writesrv -> /usr/sbin/writesrv
```

C.5 bos.compat.net Fileset

```
    Fileset    File
    ------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.net 4.1.1.0
                /usr/sbin/inetexp
                /usr/bin/inetimp -> /usr/sbin/inetimp
                /usr/sbin/notinet
                /usr/bin/inetserv -> /usr/sbin/inetserv
                /usr/sbin/inetserv
                /usr/bin/notinet -> /usr/sbin/notinet
                /usr/bin/inetexp -> /usr/sbin/inetexp
                /usr/sbin/stinet
                /usr/sbin/inetimp
                /usr/bin/stinet -> /usr/sbin/stinet
Path: /etc/objrepos
  bos.compat.net 4.1.1.0
                NONE
```

C.6 bos.compat.NetInstl Fileset

```
    Fileset    File
    ------------------------------------------------------------------------
Path: /usr/lib/objrepos
  bos.compat.NetInstl 4.1.0.0
                /usr/lpp/bosinst/bin/instsrv
                /usr/lpp/bosinst/db
                /usr/lpp/bosinst/db/sample.client
                /usr/lpp/bosinst/scripts
                /usr/lpp/bosinst/ninst
                /usr/lpp/bosinst/scripts/sendfssz
                /usr/lpp/bosinst/scripts/dbextract
                /usr/lpp/bosinst/db/choices
                /usr/lpp/bosinst/scripts/clientdb
                /usr/lpp/bosinst/scripts/curdate
```

```
                         /usr/lpp/bosinst/scripts/sendfile
                         /usr/lpp/bosinst/bin
                         /usr/lpp/bosinst/scripts/defaults
                         /usr/lpp/bosinst/scripts/netcat
                         /usr/lpp/bosinst/scripts/getdb
                         /usr/lpp/bosinst/scripts/showchoices
                         /usr/lpp/bosinst/bin/instsrv.c
                         /usr/lpp/bosinst/db/default.client
                         /usr/lpp/bosinst/db/class.sample

          Path: /etc/objrepos
            bos.compat.NetInstl 4.1.0.0
                      NONE
```

C.7 bos.compat.termcap Fileset

```
       Fileset    File
       -----------------------------------------------------------------------------
       Path: /usr/lib/objrepos
         bos.compat.termcap 4.1.1.0
                   /usr/ccs/lib/libtermcap.a
                   /usr/lib/libtermcap/termcap.src
                   /usr/lib/libtermcap.a -> /usr/ccs/lib/libtermcap.a
                   /usr/lib/libtermcap

       Path: /etc/objrepos
         bos.compat.termcap 4.1.1.0
                   /etc/termcap -> /usr/lib/libtermcap/termcap.src
```

C.8 bos.compat.termcap.data Fileset

```
       Fileset    File
       -----------------------------------------------------------------------------
       Path: /usr/share/lib/objrepos
         bos.compat.termcap.data 4.1.3.0
                   /usr/share/lib/tabset/diablo
                   /usr/share/lib/tabset/reorder
                   /usr/share/lib/tabset/std
                   /usr/share/lib/tabset/3101
                   /usr/share/lib/tabset/xerox1720
                   /usr/share/lib/tabset
                   /usr/share/lib/tabset/teleray
```

C.9 X11.compat.lib.Motif10 Fileset

```
       Fileset    File
       -----------------------------------------------------------------------------
       Path: /usr/lib/objrepos
         X11.compat.lib.Motif10 4.1.0.0
                   /usr/lpp/X11/lib/R3/uil
                   /usr/lpp/X11/lib/R3/Mrm
                   /usr/lpp/X11/lib/R3
                   /usr/lpp/X11/lib/R3/Xm
```

C.10 X11.compat.lib.Motif114 Fileset

```
      Fileset   File
      --------------------------------------------------------------------------------
Path: /usr/lib/objrepos
  X11.compat.lib.Motif114 4.1.0.0
                  /usr/lpp/X11/lib/libXm.a -> /usr/lpp/X11/lib/R4/Motif1.1.4/libXm.a
                  /usr/lpp/X11/lib/R4/Motif1.1.4
                  /usr/lpp/X11/lib/R4/Motif1.1.4/libUil.a
                  /usr/lpp/X11/lib/R4
                  /usr/lpp/X11/lib/libUil.a -> /usr/lpp/X11/lib/R4/Motif1.1.4/libUil.a
                  /usr/lpp/X11/lib/R4/Motif1.1.4/libXm.a
                  /usr/lpp/X11/lib/R4/Motif1.1.4/libMrm.a
                  /usr/lpp/X11/lib/libMrm.a -> /usr/lpp/X11/lib/R4/Motif1.1.4/libMrm.a
```

C.11 X11.compat.lib.X11R3 Fileset

```
      Fileset   File
      --------------------------------------------------------------------------------
Path: /usr/lib/objrepos
  X11.compat.lib.X11R3 4.1.0.0
                /usr/lpp/X11/lib/R3
                /usr/lpp/X11/lib/R3/X11
                /usr/lpp/X11/lib/R3/Xt
                /usr/lpp/X11/lib/R3/oldX
```

C.12 X11.compat.lib.X11R4 Fileset

```
      Fileset   File
      --------------------------------------------------------------------------------
Path: /usr/lib/objrepos
  X11.compat.lib.X11R4 4.1.0.0
                /usr/lpp/X11/lib/liboldX.a -> /usr/lpp/X11/lib/R4/liboldX.a
                /usr/lpp/X11/lib/R4/liboldX.a
                /usr/lpp/X11/lib/R4
                /usr/lpp/X11/lib/R4/libXt.a
                /usr/lpp/X11/lib/libXt.a -> /usr/lpp/X11/lib/R4/libXt.a
                /usr/lpp/X11/lib/R4/libX11.a
                /usr/lpp/X11/lib/libX11.a -> /usr/lpp/X11/lib/R4/libX11.a
```

C.13 X11.compat.samples.util Fileset

```
      Fileset   File
      --------------------------------------------------------------------------------
Path: /usr/lib/objrepos
  X11.compat.samples.util 4.1.0.0
                /usr/lpp/X11/Xamples/aixclients/aixinit/Imakefile
                /usr/lpp/X11/Xamples/defaults/xinitrc
                /usr/lpp/X11/Xamples/aixclients/aixinit/aixinit.msg
                /usr/lpp/X11/Xamples/aixclients
                /usr/lpp/X11/Xamples/bin/aixinit
                /usr/lpp/X11/Xamples/defaults
                /usr/lpp/X11/Xamples/aixclients/aixinit/aixinit.cpp
                /usr/lpp/X11/Xamples/bin
                /usr/lpp/X11/Xamples/aixclients/aixinit
```

C.14 X11.x_st_mgr.compat Fileset

```
  Fileset    File
  --------------------------------------------------------------------------
Path: /usr/lib/objrepos
  X11.x_st_mgr.compat 4.1.0.0
            /usr/lpp/x_st_mgr/bin/xset
            /usr/lpp/x_st_mgr/bin/conf_tab
            /usr/lpp/x_st_mgr/bin/start_server
```

Appendix D. rmlpp Shell Script

Following is the source of the rmlpp script:

```
#!/bin/ksh
# Permanently remove a product from disk and AIX databases
# Updated on 9 January 1993 to include full paths on each command,
# and to use rmdir on directories so errant LPP inventory information
# will not cause the wrong files to be removed.  To get the previous
# behavior, see the comment below about the RMLPP_BRUTE_FORCE_REMOVE
# variable.                 -MAC
if [ `/bin/whoami` != "root" ] ; then
  echo You must be root to run this script.
  exit 1
fi
if [ `/bin/uname -a | /bin/awk '{print $1$4$3}'` != "AIX32" ] ; then
  echo This script only works on AIX 3.2.
  exit 1
fi
TMP_FREE=`/bin/df /tmp | /bin/awk '$3 ~ /[0-9]/{print $3}'`
if [ "$TMP_FREE" -lt 1000 ] ; then
    echo There is not enough room in your /tmp directory.
    echo You need 1000 KB free, and you have only $TMP_FREE KB free.
    echo Either remove some stuff from /tmp, or use chfs to make it bigger.
    exit 1
fi
ODMDIRS="/etc/objrepos /usr/lib/objrepos /usr/share/lib/objrepos"
ODMDIR=/usr/lib/objrepos
export ODMDIR
if [ $# -lt 1 ]
then
  echo usage: $0 lppname [lppname ...]
  echo lppname is a string compatible with grep, ie "X11" or "PHIGS"
  echo typing   $0 PHIGS   will remove all LPPs with PHIGS in their name.
  exit 1
fi
NAMES=$1
shift
while [ $# -gt 0 ] ; do
  NAMES="$NAMES|$1"
  shift
done
echo "Searching for lpps with egrep \"$NAMES\"...\c"
for ODMDIR in $ODMDIRS ; do
  if [ ! -d $ODMDIR -o ! -w $ODMDIR ] ; then
      echo $ODMDIR is not writeable or is not a directory.
      echo I hope this is because you are a /usr client or diskless.
      echo If you are not a diskless or a /usr client, you should stop.
      echo "Enter y to continue ->\c"
      read answer
      if [ "$answer" != "y" ] ; then
        exit 0
      fi
  fi
  TMP=`/bin/odmget lpp |/bin/awk -F\" '/name/ {print $2}' |/bin/egrep "$NAMES"`
  LPPS=`echo $LPPS $TMP`
done
```

```
if [ "$LPPS" = "" ]
then
  echo "failed.\nNo LPP with the name $NAMES detected."
  exit 1
fi
echo ok.
for ODMDIR in $ODMDIRS ; do
  /bin/mkdir -p /tmp/rmlpp/$ODMDIR > /dev/null 2>&1
done
> /tmp/listOfFilesToRM$$    # truncate temporary file, just in case.
echo
echo This script is about to attempt to remove an LPP from your system.
echo I say attempt, because it could fail.  If it fails, you may have
echo to at least reload the LPP.  Use the \"lppchk\" command to make sure
echo all is well with your system.
echo
# Loop through all the LPP names found.
for LPP in $LPPS; do
  DESCR=none
  answer=""
  # find the LPP ids.  They will be different in the three SWVPD databases.
  for ODMDIR in $ODMDIRS ; do
    # get the lpp id for this ODMDIR (yes, they are different)
    LPPID=`/bin/odmget -qname=$LPP lpp |/bin/grep lpp_id |/bin/sed 's/.* = //'`
    # did we find the LPP?
    if [ "$DESCR" = "none" -a "$LPPID" != ""  ] ; then
      # all the descriptions should be the same
      DESCR=`/bin/odmget -qname=$LPP lpp |/bin/grep description | \
            /bin/sed 's/.* = //'`
      echo "Delete $LPP, $DESCR?"
      echo "y or (n) ->\c"
      read answer
      if [ "$answer" != "y" ] ; then # jump back up to the next LPP name
        continue 2
      fi
    fi
    # if there is no DESCR, then we didn't find the LPP.  Weird.
    if [ "$DESCR" = "none" -o "$LPPID" = "" ] ; then
      continue
    fi
        SOMETHING_DONE_FLAG=true
    # Optionally, save the ODM stuff we are about to remove,
    # in case something goes wrong.  The problem is it is difficult
    # to determine if something really did fail, since these commands
    # don't return any decent error return codes.
    /bin/odmget -q "lpp_id=$LPPID" history  >/tmp/rmlpp/$ODMDIR/$LPP.history
    /bin/odmget -q "name=$LPP" lpp          >/tmp/rmlpp/$ODMDIR/$LPP.lpp
    /bin/odmget -q "lpp_name=$LPP" product  >/tmp/rmlpp/$ODMDIR/$LPP.product
    # Get the list of files and links to remove later....
    /bin/odmget -q "lpp_id=$LPPID" inventory > /tmp/rmlpp/$ODMDIR/$LPP.inventory
    /bin/awk -F\" '/loc/ {print $2}' /tmp/rmlpp/$ODMDIR/$LPP.inventory | \
      /bin/sed 's/,/ /g' >> /tmp/listOfFilesToRM$$
    /bin/odmdelete -o history -q lpp_id=$LPPID  > /dev/null 2>&1
    /bin/odmdelete -o lpp -q name=$LPP          > /dev/null 2>&1
    /bin/odmdelete -o product -q lpp_name=$LPP  > /dev/null 2>&1
    /bin/odmdelete -o inventory -q lpp_id=$LPPID > /dev/null 2>&1
```

```
    done
done
if [ "$SOMETHING_DONE_FLAG" = "true" ] ; then
  echo ODM work is done.  Now, time to delete files....
  # This could be catastrophic if there is a problem.  For example,
  # if the ODM database for an application had / as one of its files.
  # You be the judge.  Here's your rope....
  # cat /tmp/listOfFilesToRM$$ | sort -r | uniq | xargs rm -rf
  # I decided to change this to use rm on files, and rmdir on
  # directories.  This is done because some LPPs do not add
  # inventory correctly, so it will reduce the chance of disaster.
  #                                     -MAC    9 January 1993
  /bin/cat /tmp/listOfFilesToRM$$ | /bin/sort -r | /bin/uniq | while read target
  do
    if [ -L "$target" ] ; then   # Symbolic link is "special".
      /bin/rm -f $target
    elif [ -d "$target" ] ; then
      /bin/rm -r $target
    else
      /bin/rm -f $target
    fi
  done
fi
  # If you want the old behavior, where rmlpp brutally removes all
  # files and directories that the LPP has said he owns, then you
  # can simply define the RMLPP_BRUTE_FORCE_REMOVE environment variable.
  # Keep in mind that at least two LPPs I know of (xlCbrs.obj and
  # bsl.ja_JP.com.loc) include BAD inventory information, and removing
  # these LPPs with the RMLPP_BRUTE_FORCE_REMOVE flag set will cause
  # serious problems with your system (like a reinstall).
  # RMLPP_BRUTE_FORCE_REMOVE=DangerousOption
  if [ -n "$RMLPP_BRUTE_FORCE_REMOVE" ] ; then
    cat /tmp/listOfFilesToRM$$ | sort -r | uniq | xargs rm -rf
  fi
  /bin/rm -rf /tmp/listOfFilesToRM$$
  echo done.
# take this line out if you want to save your ODM saved files.
/bin/rm -rf /tmp/rmlpp
exit 0
```

Appendix E. AIX V3.2 to AIX V4.1 Fileset Mapping

Table 28 shows the mapping of the AIX V3.2 filesets to AIX V4.1 filesets that takes place as part of a Migration Installation to AIX V4. If the AIX V3.2 fileset in the left column is installed on the system, the equivalent AIX V4.1 fileset(s) in the right column are installed.

Table 28 (Page 1 of 4). AIX V4 Migration Install - Fileset Mapping	
AIX V3.2 Fileset Name	**AIX V4.1 Fileset Name**
blkmux.obj	devices.mca.fe92.rte
bos.data	bos.data
bos.obj	bos.compat.cmds bos.compat.links bos.compat.NetInstl bos.sysmgt.quota bos.sysmgt.serv_aid bos.sysmgt.smit bos.sysmgt.sysbr bos.sysmgt.trace bos.compat.termcap bos.iconv.com printers.rte bos.info.rte
bosadt.bosadt.data	bos.adt.data
bosadt.bosadt.obj	bos.adt.base bos.adt.sccs bos.adt.debug bos.adt.utils
bosadt.lib.obj	bos.adt.include bos.adt.lib bos.compat.libs
bosadt.lib.obj	bos.adt.libm bos.net.nfs.adt bos.adt.syscalls
bosadt.prof.obj	bos.adt.prof
bosext1.mh.obj	bos.mh
bosext1.uucp.obj	bos.net.uucp
bosext2.acct.obj	bos.acct
bosext2.ate.obj	bos.net.ate
bosext2.dosutil.obj	bos.dosutil
bosext2.games.obj	bos.games
bosext2.lrn.data	bos.lrn
bosnet.ncs.obj	bos.net.ncs
bosnet.nfs.obj	bos.net.nfs bos.net.nis
bosnet.snmpd.obj	bos.net.tcp.adt bos.net.tcp.server

Table 28 (Page 2 of 4). AIX V4 Migration Install - Fileset Mapping	
AIX V3.2 Fileset Name	**AIX V4.1 Fileset Name**
bosnet.tcpip.obj	bos.net.tcp bos.compat.net
bosrcs.obj	bos.rcs
bsl.aix31.imk.loc	bos.compat.imk
bsl.da_DK.aix.loc	bos.loc.iso.da_DK
bsl.da_DK.pc.loc	bos.loc.pc_compat.Da_DK
bsl.de_CH.aix.loc	bos.loc.iso.de_CH
bsl.de_CH.pc.loc	bos.loc.pc_compat.De_CH
bsl.de_DE.aix.loc	bos.loc.iso.de_DE
bsl.de_DE.pc.loc	bos.loc.pc_compat.De_DE
bsl.el_GR.aix.loc	bos.loc.iso.el_GR bos.loc.fnt.iso7
bsl.en_GB.aix.loc	bos.loc.iso.en_GB
bsl.en_GB.pc.loc	bos.loc.pc_compat.En_GB
bsl.en_US.aix.loc	bos.loc.iso.en_US
bsl.en_US.pc.loc	bos.loc.pc_compat.En_US
bsl.es_ES.aix.loc	bos.loc.iso.es_ES
bsl.es_ES.pc.loc	bos.loc.pc_compat.Es_ES
bsl.fi_FI.aix.loc	bos.loc.iso.fi_FI
bsl.fi_FI.pc.loc	bos.loc.pc_compat.Fi_FI
bsl.fr_BE.aix.loc	bos.loc.iso.fr_BE
bsl.fr_BE.pc.loc	bos.loc.pc_compat.Fr_BE
bsl.fr_CA.aix.loc	bos.loc.iso.fr_CA
bsl.fr_CA.pc.loc	bos.loc.pc_compat.Fr_CA
bsl.fr_CH.aix.loc	bos.loc.iso.fr_CH
bsl.fr_CH.pc.loc	bos.loc.pc_compat.Fr_CH
bsl.fr_FR.aix.loc	bos.loc.iso.fr_FR
bsl.fr_FR.pc.loc	bos.loc.pc_compat.Fr_FR
bsl.greek.fnt.loc	bos.loc.fnt.iso7
bsl.is_IS.aix.loc	bos.loc.iso.is_IS
bsl.is_IS.pc.loc	bos.loc.pc_compat.Is_IS
bsl.it_IT.aix.loc	bos.loc.iso.it_IT
bsl.it_IT.pc.loc	bos.loc.pc_compat.It_IT
bsl.ja_JP.aix.loc	bos.loc.iso.ja_JP
bsl.ja_JP.com.loc	bos.loc.com.JP
bsl.ja_JP.pc.loc	bos.loc.pc.Ja_JP
bsl.ko_KR.aix.loc	bos.loc.iso.ko_KR
bsl.lat-1.fnt.loc	bos.loc.fnt.iso1 bos.loc.com.EE

Table 28 (Page 3 of 4). AIX V4 Migration Install - Fileset Mapping	
AIX V3.2 Fileset Name	**AIX V4.1 Fileset Name**
bsl.nl_BE.aix.loc	bos.loc.iso.nl_BE
bsl.nl_BE.pc.loc	bos.loc.pc_compat.Nl_BE
bsl.nl_NL.aix.loc	bos.loc.iso.nl_NL
bsl.nl_NL.pc.loc	bos.loc.pc_compat.Nl_NL
bsl.no_NO.aix.loc	bos.loc.iso.no_NO
bsl.no_NO.pc.loc	bos.loc.pc_compat.No_NO
bsl.pt_PT.aix.loc	bos.loc.iso.pt_PT
bsl.pt_PT.pc.loc	bos.loc.pc_compat.Pt_PT
bsl.sv_SE.aix.loc	bos.loc.iso.sv_SE
bsl.sv_SE.pc.loc	bos.loc.pc_compat.Sv_SE
bsl.tr_TR.aix.loc	bos.loc.iso.tr_TR bos.loc.fnt.iso9
bsl.turki.fnt.loc	bos.loc.fnt.iso9
bsl.zh_TW.aix.loc	bos.loc.iso.zh_TW
fddi.obj	devices.mca.8ef4.com
INed.obj	bos.INed
pci.obj	bos.pci
txtfmt.bib.data	bos.txt.bib.data
txtfmt.bib.obj	bos.txt.bib
txtfmt.hplj.fnt	bos.txt.hplj.fnt
txtfmt.ibm3812.fnt	bos.txt.ibm3812.fnt
txtfmt.ibm3816.fnt	bos.txt.ibm3816.fnt
txtfmt.spell.data	bos.txt.spell.data
txtfmt.spell.obj	bos.txt.spell
txtfmt.tfs.data	bos.txt.tfs.data
txtfmt.tfs.obj	bos.txt.tfs
txtfmt.ts.obj	bos.txt.ts
txtfmt.xpv.obj	bos.txt.xpv.rte
X11dev.im	X11.samples.im
X11dev.motif1.2.obj	X11.adt.motif
X11dev.motif1.2.src	X11.samples.apps.motifdemos
X11dev.obj	X11.adt bos.adt.graphics
X11dev.src	X11.samples X11.adt.imake
X11fnt.coreX.fnt	X11.fnt.coreX
X11fnt.ibm850.pc.fnt	X11.compat.fnt.pc
X11fnt.iso88591.aix.fnt	X11.fnt.iso1
X11fnt.iso88592.fnt	X11.fnt.iso2

Table 28 (Page 4 of 4). AIX V4 Migration Install - Fileset Mapping	
AIX V3.2 Fileset Name	**AIX V4.1 Fileset Name**
X11fnt.iso88593.fnt	X11.fnt.iso3
X11fnt.iso88594.fnt	X11.fnt.iso4
X11fnt.iso88595.fnt	X11.fnt.iso5
X11fnt.iso88597.aix.fnt	X11.fnt.iso7
X11fnt.iso88599.aix.fnt	X11.fnt.iso9
X11fnt.oldX.fnt	X11.compat.fnt.oldX
X11rte.ext.obj	X11.info.rte X11.fnt.util
X11rte.motif1.2.obj	X11.motif.lib
X11rte.obj	X11.apps X11.base X11.compat X11.motif X11.fnt.defaultFonts
X11vsm.vsm.obj	X11.vsm.rte
xlCrte.obj	xlC.rte
x_st_mgr.obj	X11.x_st_mgr

Appendix F. Device Configuration Recovery Files

This section gives the content of some shell scripts and configuration files used to recover device configurations.

F.1 fbcheck Script

Following is the source of the /usr/sbin/fbcheck shell script:

```
#!/bin/ksh
# @(#)04  1.7  src/bos/usr/sbin/fbcheck/fbcheck.sh, bosboot, bos411, 9439C411e 9/30/94
#
#    COMPONENT_NAME: BOSBOOT
#
#    FUNCTIONS: none
#
#    ORIGINS: 27
#
#
#    (C) COPYRIGHT International Business Machines Corp. 1989,1993
#    All Rights Reserved
#    Licensed Materials - Property of IBM
#    US Government Users Restricted Rights - Use, duplication or
#    disclosure restricted by GSA ADP Schedule Contract with IBM Corp.
#
#------------------------------ fbcheck --------------------------------------
#
# NAME: fbcheck
#
# FUNCTION:
# if the /etc/firstboot script exists, this program renames it and executes it
# the /etc/firstboot script is the mechanism which is used during the
# system's first boot to change a client's environment
#
# For Online software license display, this program will invoke the view and sign of
# the license agreement if the current license has been signed with a 0000. This is used
# to allow the system to be set up preserving the capability for the end customer to vie
# and sign the license.
#
# EXECUTION ENVIRONMENT:
# this file should ALWAYS have an entry in the /etc/inittab file, so that
# changes made with the  customization commands will be made the next time
# the client boots
#
# NOTES:
# refer to the corresponding man page or use InfoExplorer to obtain
# detailed information about this command
#
#

FBNAME=/etc/fb_`date +"%H_%M_%m_%d"`

if [ -s "/etc/firstboot" ]
then
 # make sure it's executable
 chmod +x /etc/firstboot
```

```
                    # rename this file so that it's not executed again
                    mv /etc/firstboot $FBNAME

                    # execute this shell script
                    $FBNAME
                 fi
                 if [ -s "/.license.sig.tmp"  ] ; then
                  NAME=`cat /.license.sig.tmp`
                  if [ "$NAME" = "0000" ] ; then
                     /var/adm/.license.sig
                   export SOFTLIC_ONLY=1
                   /usr/sbin/install_assist
                  fi
                 fi
```

F.2 firstboot Shell Script

Following is the source of the /etc/firstboot shell script:

```
/usr/lpp/bosinst/rda  -d /tmp/bos/objrepos.inst/CuDv.sav \
                      -a /tmp/bos/objrepos.inst/CuAt.sav \
                      -s /tmp/reconfig1 >/dev/null 2>&1
/usr/lpp/bosinst/rda  -d /tmp/bos/objrepos.inst/CuDv.sav \
                      -a /tmp/bos/objrepos.inst/CuAt.sav \
                      -s /tmp/reconfig2 >/dev/null 2>&1
/usr/lpp/bosinst/rda  -d /tmp/bos/objrepos.inst/CuDv.sav \
                      -a /tmp/bos/objrepos.inst/CuAt.sav \
                      -s /tmp/reconfig3 >/dev/null 2>&1

cfgmgr
```

F.3 rda Command

Following is the syntax of the rda command:

```
 rda [-a attr_file_name][-c prefix][-d dvc_file_name][-p object path] \
     [-s scriptname ][-eRrx]
   where
    -a : specifies the name of output attribute stanza file;
         default is "/etc/objrepos/old_ca_stanzas".
    -c : prefix to search for in the old CuDv;
         default is "'*'".
    -d : specifies the  name of output device stanza file;
         default is "/etc/objrepos/old_cd_stanzas".
    -p : specifies directory pathname containing the old object classes;
         default is "/etc/objrepos".
    -e : extracts only from the old database and does not recover the
         device or configuration information and does not execute the
         recovery script.
    -R : Relaxes the criteria for determining which device to configure.
         If this flag is set, the location code will not be checked.
    -r : recovers only device and configuration information from the
         stanza files.
         Does not extract information from the old database and
         does not execute the recovery script.
    -s : name of shell script file to use for preserving data.
    -x : executes only the script containing recovery config methods.
```

F.4 Recovery Shell Scripts

Following is the source of the /tmp/reconfig1 recovery shell script:

```
#!/bin/ksh
export TALLY=
cmd_0='/etc/methods/define -l tty0 -c tty -s rs232 -t tty \
                                        -p sa1 -w s2 '
cmd_1='/etc/methods/cfgtty -l tty0 '
cmd_2='/usr/lib/methods/define -l lp0 -c printer -s parallel \
                                      -t ibm4216-31 -p ppa0 -w p '
cmd_3='/usr/lib/methods/cfgpp -l lp0 '
cmd_4='chdev -l sys0
 -a maxuproc=100'
cmd_5='chdev -l tok0
 -a ring_speed=16'
cmd_6='chdev -l tty0
 -a term=ibm3151
 -a login=enable'
cmd_7='chdev -l inet0
 -a hostname=migserve.eth.itsc.austin.ibm.com
 -a route=net,,0,9.3.1.74'
cmd_8='chdev -l en0
 -a state=up
 -a netaddr=9.3.2.28
 -a netmask=255.255.255.0'
cmd_9='chdev -l tr0
 -a state=up
 -a netaddr=9.3.1.28
 -a netmask=255.255.255.0'
num_cmds=10
i=0
while [ $i -lt $num_cmds ]
do
 eval \$cmd_$i > /dev/null
 if [ $? -ne 0 ]
 then
  TALLY="$TALLY$(eval echo \$cmd_$i)\n"
 fi
 (( i+=1 ))
done
if [ -n "$TALLY" ]
then
 echo Machine not identical to previous configuration.
 echo The following items failed:
 echo $TALLY
fi
```

Following is the source of the /tmp/reconfig2 recovery shell script:

```
#!/bin/ksh
export TALLY=
cmd_0='chdev -l sys0
 -a maxuproc=100'
cmd_1='chdev -l tok0
 -a ring_speed=16'
cmd_2='chdev -l tty0
 -a term=ibm3151
```

```
                -a login=enable'
cmd_3='chdev -l inet0
 -a hostname=migserve.eth.itsc.austin.ibm.com
 -a route=net,,0,9.3.1.74'
cmd_4='chdev -l en0
 -a state=up
 -a netaddr=9.3.2.28
 -a netmask=255.255.255.0'
cmd_5='chdev -l tr0
 -a state=up
 -a netaddr=9.3.1.28
 -a netmask=255.255.255.0'
num_cmds=6
i=0
while [ $i -lt $num_cmds ]
do
 eval \$cmd_$i > /dev/null
 if [ $? -ne 0 ]
 then
  TALLY="$TALLY$(eval echo \$cmd_$i)\n"
 fi
 (( i+=1 ))
done
if [ -n "$TALLY" ]
then
 echo Machine not identical to previous configuration.
 echo The following items failed:
 echo $TALLY
fi
```

Appendix G. AIX V4.1 Device Driver Packages

Following is a list of AIX Version 4.1.4 device driver packages and their description. This list should help you in retrieving the name of a device driver package (BFF image) that is needed to support a specific hardware.

Note: In general, each package listed below contains several filesets, such as the device driver code itself (.rte), the diagnostics (.diag), the microcode for the hardware (.ucode), and others. These filesets are not listed here.

G.1.1 Base Device Drivers

devices.base	Common and Uniprocessor Base System Device Software
devices.rs6ksmp.base	Multiprocessor Base System Device Software
devices.mca.dee6	Standard I/O (dee6) Adapter Software
devices.mca.df5f	Standard I/O (df5f) Adapter Software
devices.mca.df9f	Direct Attached Disk Software
devices.mca.fed9	Standard I/O (fed9) Adapter Software
devices.mca.f6fe	Standard I/O (f6fe) Adapter Software
devices.mca.f6f8	MCA Keyboard & Mouse Adapter (f6f8) Software
devices.mca.8d77	8-bit SCSI I/O Controller Software
devices.mca.8efc	16-bit SCSI I/O Controller Software
devices.mca.8fba	Standard NCR53C720 SCSI Software
devices.mca.8ff4	Standard NCR53C700 SCSI Software
devices.mca.8f78	Serial Linked Disk Adapter/Controller/DASD Software
devices.mca.8f97	SSA Adapter (8f97) Software
devices.mca.8f9d	LAN SCSI Adapter
devices.scsi.disk	SCSI CD-ROM, Disk, Read/Write Optical Device Software
devices.scsi.scarray	RAIDiant Array DA Device Software Support
devices.scsi.tape	SCSI Tape Device Software
devices.scsi.tm	SCSI Target Mode Software
devices.sio.fda	Diskette Adapter Software
devices.sio.ktma	Keyboard Tablet & Mouse Device and Adapter Software
devices.sio.ppa	Parallel Printer Adapter Software
devices.ssa.disk	SSA DASD Software
devices.sys.mca	Micro Channel Bus Software

319

G.1.2 TTY Device Drivers

devices.sio.sa	Built-in Serial Adapter Software
devices.tty	TTY Device Driver Support Software
devices.7318.com	7318 Common Download Software
devices.7318.p10	7318 model P10 Device Driver Support
devices.7318.s20des	7318 model S20 Download Software w/DES En cryption
devices.7318.s20	7318 model S20 Host Utility Software
devices.mca.edd0	8-Port Asynchronous Adapter EIA-232 Software
devices.mca.edd1	8-Port Asynchronous Adapter EIA-422 Software
devices.mca.edd2	8-Port Asynchronous Adapter MIL-STD 188 Software
devices.mca.edd3	16-Port Asynchronous Adapter EIA-422 Software
devices.mca.edd6	16-Port Asynchronous Adapter EIA-232 Software
devices.mca.61fd	64-Port Asynchronous Adapter Software
devices.mca.ffe1	128-Port Asynchronous Adapter Software

G.1.3 Communication Device Drivers

devices.mca.deff	Multiprotocol Adapter/A (MP/A) Configuration Support
devices.mca.eff0	X.25 CoProcessor/2 or Multiport/2 Adapter Software
devices.mca.e1ff	3270 Connection Adapter Software
devices.mca.fe92	BLKMUX Adapter (fe92) Software
devices.mca.8ef2	Integrated Ethernet Adapter (8ef2) Software
devices.mca.8ef3	Integrated Ethernet Adapter (8ef3) Software
devices.mca.8ef4	FDDI Adapter (8ef4) Software
devices.mca.8ef5	Ethernet High-Performance LAN Adapter (8ef5) Software
devices.mca.8fa2	Token-Ring High-Performance Adapter (8fa2) Software
devices.mca.8fc3	ESCON Adapter (8fc3) Software
devices.mca.8fc8	Token-Ring High-Performance Adapter (8fc8) Software
devices.mca.8fe2	FCS Adapter (8fe2) Software
devices.mca.8fe2.ucode	STAR Adapter Microcode
devices.mca.8fe5	HIPPI Adapter Device Driver
devices.mca.8f67	Turboways 155 MCA ATM Adapter (8f67) Software
devices.mca.8f7f	Turboways 100 MCA ATM Adapter (8f7f) Software
devices.mca.8f70	4-Port Selectable Portmaster Adapter/A (MPQP) Software
devices.mca.8f70.client	4-Port Selectable Portmaster Adapter/A (MPQP) Software

devices.mca.8f95	Ethernet High-Performance LAN Adapter (8f95) Software
devices.mca.8f98	Integrated Ethernet Adapter (8f98) Software
devices.sys.slc	Serial Optical Link Software

G.1.4 Graphics and Audio/Video Device Drivers

devices.buc.00004007	GXT155L Graphics Adapter Software
devices.buc.00004006	GXT150L Graphics Adapter Software
devices.buc.00004005	GXT150 Graphics Adapter Software
devices.buc.00004002	GXT500 Graphics Adapter Software
devices.buc.00004001	GXT100 Graphics Adapter Software
devices.graphics	Graphics Adapter Common Software plus VOO and Stereo Software
devices.mca.dfe5	AIX Ultimedia Services RISC Audio Device
devices.mca.edd5	Graphics Input Adapter Software
devices.mca.efc6	BBLMCA Graphics Adapter Software
devices.mca.e555	Digital Trunk Adapter (e555) Software
devices.mca.e556	Digital Trunk Dual Adapter (e556) Software
devices.mca.8ee3	Gt3/Gt4 Graphics Adapter Software
devices.mca.8ee4	Color Graphics Display Adapter Software
devices.mca.8ee5	Grayscale Graphics Display Adapter Software
devices.mca.8fbc	GXT1000 Graphics Adapter Software
devices.mca.8ffd	High-Speed 3D Graphics Adapter Software
devices.mca.8f9a	GXT150M Graphics Adapter Software
devices.mca.8f96	AIX Ultimedia Services Video I/O Adapter
devices.mca.8787	5085/86/88 Attachment Adapter Software
devices.serial.gio	Serial Graphics Input Adapter Software
devices.serial.sb1	6094-030 Spaceball 3-D Input Device Software
devices.serial.tablet1	Serial Tablet Input Device Software
devices.sys.sga	Gt1 Graphics Adapter Software
devices.sys.sgabus	Special Graphics Slot Software
devices.sys.wga	Gt1x Graphics Adapter Software

G.1.5 RSPC Device Drivers

devices.rspc.base	RISC PC Base System Device Software
devices.scsi.disk	RISC PC SCSI CD-ROM, Disk, Read/Write Optical Device Software
devices.scsi.tape	RISC PC SCSI Tape Device Software
devices.sys.pci	PCI Bus Software

devices.sys.PNP0A03	PCI Bus Software (PNP0A03)
devices.ide.cdrom	IDE CDROM Device Software
devices.ide.disk	IDE Disk Device Software
devices.isa.cxia128	ISA 128-Port Asynchronous Adapter Software
devices.isa.cxia	ISA 8-Port Asynchronous Adapter Software
devices.isa.c1x	X.25 CoProcessor/1 Adapter Software
devices.isa.ethernet	IBM ISA Ethernet Adapter Software
devices.isa.mm2	Multiport Model 2 Adapter Software
devices.isa.pcmcia	PCMCIA Bus Adapter Software
devices.isa.sio	RISC PC Standard I/O Software
devices.isa.tokenring	IBM 16/4 PowerPC Token-Ring Adapter (isa) Software
devices.isa.IBM0010	ISA Ethernet Adapter Software (IBM0010)
devices.isa.PNP80CC	IBM 16/4 PowerPC Token-Ring Adapter (isa-PNP80CC) Software
devices.isa_sio.sa	RISC PC Standard Serial Adapter Software
devices.isa_sio.ppa	RISC PC Parallel Printer Adapter Software
devices.isa_sio.km	ISA Keyboard & Mouse Software
devices.isa_sio.baud.NA	Empty Fileset for UMS LPP Audio Dev. Support
devices.isa_sio.baud	AIX Ultimedia Services New RISC PC Audio Device
devices.isa_sio.PNP0F03	ISA Mouse Software (PNP0F03)
devices.isa_sio.PNP0E00	PCMCIA Bus Software (PNP0E00)
devices.isa_sio.PNP0700	RISC PC Diskette Adapter Software (PNP0700)
devices.isa_sio.PNP0600	IDE Adapter Device Software
devices.isa_sio.PNP0501	RISC PC Standard Serial Adapter Software (PNP0501)
devices.isa_sio.PNP0401	RISC PC ECP Parallel Port Adapter Software (PNP0401)
devices.isa_sio.PNP0400	RISC PC Standard Parallel Port Adapter Software(PNP0400)
devices.isa_sio.PNP0303	ISA Keyboard Software (PNP0303)
devices.isa_sio.fda	RISC PC Diskette Adapter Software
devices.isa_sio.IBM001D.NA	Empty Fileset for Optional Yamaha Audio Support
devices.isa_sio.IBM001C	RISC PC EPP Parallel Port Adapter Software (IBM001C)
devices.isa_sio.IBM0017.NA	Empty Fileset for UMS LPP Audio Dev. Support
devices.isa_sio.IBM0017	AIX Ultimedia Services RISC PC Audio Device (CS4232)
devices.isa_sio.IBM0013.NA	Empty Fileset for Optional CS4232 MIDI Support
devices.isa_sio.IBM0012.NA	Empty Fileset for Optional CS4232 Gameport

devices.isa_sio.IBM000E.NA Empty Fileset for UMS LPP Audio Dev. Support

devices.isa_sio.IBM000E AIX Ultimedia Services RISC PC Audio Device (CS4231)

devices.isa_sio.IBM0005.IBM8301 ISA Power Management Controller Software

devices.pci.ed101073 G10 Graphics with Motion Video Adapter Software

devices.pci.PNP0A03 PCI Bus Bridge Software

devices.pci.PNP0A00 ISA Bus Bridge Software (PNP0A00)

devices.pci.0e100091 H10/S15 Graphics Adapter Software

devices.pci.00100300 PCI 16-bit SCSI I/O Controller Software

devices.pci.00100100 Standard NCR53C810 SCSI Software

devices.pci.1c104ac2 G10 Graphics Adapter Software

devices.pci.14102000 GXT1000 PCI Graphics Adapter Software

devices.pci.14101c00 Power Management Controller Software

devices.pci.14101b00 GXT150P Graphics Adapter Software

devices.pci.14101800 IBM PCI Token-Ring Adapter Software

devices.pci.14101300 S10 Graphics Adapter Software

devices.pci.22100020 IBM PCI Ethernet Adapter Software

devices.pci.3353c388 E15 Graphics Adapter Software 3

devices.pci.3353c288 E15 Graphics Adapter Software 2

devices.pci.3353c188 E15 Graphics Adapter Software 1

devices.pci.3353c088 E15 Graphics Adapter Software 0

devices.pci.3353b088 E10 Graphics Adapter Software

devices.pci.33531188 E15 Graphics Adapter Software 4

devices.pci.86808404 ISA Bus Software

devices.pcmcia.a4001e00 IBM PCMCIA Token-Ring Card Software

devices.pcmcia.a4001d00 IBM PCMCIA High-Speed Fax Modem Card Software

devices.pcmcia.a4000200 IBM PCMCIA Ethernet Card Software

devices.pcmcia.ethernet PCMCIA Ethernet Card Software

devices.pcmcia.serial PCMCIA Serial Port Software

devices.pcmcia.tokenring PCMCIA Token-Ring Card Software

List of Abbreviations

2xx	Represents any RISC System/6000 model with a model number starting with 2. For example, 220 or 25T.
3xx	Represents any RISC System/6000 model with a model number starting with 3. For example, 350 or 39H.
5xx	Represents any RISC System/6000 model with a model number starting with 5. For example, 580 or 591.
9xx	Represents any RISC System/6000 model with a model number starting with 9. For example, 98B or 990.
ACL	Access Control List
ACPA	Audio Capture and Playback Adapter
ADK	Application Developer's Toolkit
AIX	Advanced Interactive eXecutive
AIX V3.2	AIX Version 3.2 Operating System
AIX V4.1	AIX Version 4.1 Operating System
ADSM	Adstar Distributed Storage Manager
ALP	Agreement for IBM Licensed Programs
ARK	Adminstrator Runtime Kit
ARTIC	A Real Time Interface Coprocessor
ASCII	American Standard Code for Information Interchange
BFF	Backup File Format
BMUX	Block MUltipleXer
BOS	Base Operating System
Cxx	Represents any RISC System/6000 model with a model number starting with C. For example, C10 or C20.
CD	Compact Disc
CDE	Common Desktop Environment
CD-ROM	Compact Disc Read Only Memory
COU	Conditions Of Use

CPU	Central Processing Unit
DCE	Distributed Computing Environment
DES	Data Encryption Standard
DNS	Domain Name System
DOS	Disk Operating System
DSLO	Distributed Systems License Option
ESCON	Enterprise Systems Connection
FDDI	Fibre Distributed Data Interface
GA	General Availability
GLB	Global Location Broker
GUI	Graphical User Interface
Gxx	Represents any RISC System/6000 model with a model number starting with G. For example, G30.
HACMP	High Availability Cluster MultiProcessing
HFT	High-Function Terminal
HP	Hewlett-Packard
IBM	International Business Machines Corporation
iFOR/LS	information For Operation and Retrieval/License System
IPC	Inter Process Communication
IPL	Initial Program Load
IPLA	International Program License Agreement
IPX	Internetwork Package eXchange
ISV	Independent Software Vendor
ITSO	International Technical Support Organization
JFS	Journalled File System
Jxx	Represents any RISC System/6000 model with a model number starting with J. For example, J30.
LAN	Local Area Network
LCD	Liquid Crystal Display
LED	Light Emitting Diode

LFT	Low-Function Terminal	**RISC**	Reduced Instruction Set Computer or Reduced Instruction Set Cycles
LPP	Licensed Program Product		
LVCB	Logical Volume Control Block	**RSC**	RISC Single Chip
LVM	Logical Volume Manager	**RS/6000**	IBM RISC System/6000
MP	Multiprocessor	**RS/6000 SP**	IBM RISC System/6000 Scalable POWERParallel
MPA	Multi-Protocol Adapter		
MTU	Maximum Transmission Unit	**Rxx**	Represents any RISC System/6000 model with a model number starting with R. For example, R24 or R30.
NCK	Network Computing Kernel		
NCS	Network Computing System		
NetBIOS	Network Basic Input Output System	**SCSI**	Small Computer Systems Interface
NFS	Network File System	**SID**	System Identifier
NIM	Network Installation Manager	**SIMM**	Serial In-line Memory Module
NLS	National Language Support	**SMIT**	System Management Interface Tool
NT	New Technology		
NTA	Network Terminal Accelerator	**SMP**	Symetric Multiprocessor
NW	NetWare	**SNA**	Systems Network Architecture
ODM	Object Data Manager	**SP1**	IBM Scalable POWERParallel Systems 1
OFA	Optimal Flexible Architecture		
OPP	Optional Program Product	**SP2**	IBM Scalable POWERParallel Systems 2
OS	Operating System		
OS/2	Operating System/2	**SPX**	Sequenced Packet eXchange Transport Protocol
PC	Personal Computer		
PCI	Peripheral Component Interconnect	**SPX2**	Enhanced Sequenced Packet eXchange Transport Protocol
PMP	Preventive Maintenance Package	**TCB**	Trusted Computing Base
		TCP/IP	Transmission Control Protocol / Internet Protocol
POSIX	Portable Operating System Interface for Computer Environments	**TTY**	TeleType or asynchronous terminal
POST	Power-On Self-Test	**UDP**	Unreliable Datagram Protocol
POWER	Performance Optimization With Enhanced RISC	**UNIX**	UNIX Operating System
		UK	United Kingdom
PP	Physical Partitions	**UP**	UniProcessor
PTF	Program Temporary Fix	**US**	United States of America
RAM	Random Access Memory	**UUCP**	Unix to Unix Copy Program
RAN	Remote Asynchronous Node	**UUID**	Universal Unique Identifier
RAS	Reliability, Availability, Serviceability	**VCA**	Video Capture Adapter
		VGID	Volume Group Identifier
RDBMS	Relational Database Management System	**VGDA**	Volume Group Descriptor Area
		VPD	Vital Product Data
RFC	Request For Comment	**VRMF**	Version, Release, Modification, Fix
		VSM	Visual Systems Manager

WDFM	Withdrawn From Marketing	**XPG4**	X/Open Portability Guide, Issue 4
WWW	World Wide Web		

Index

Special Characters

.oldvgs file   118, 138

Numerics

000   101, 113
111   101, 113
333   101, 113

A

abbreviations   325
access control lists   17
ACLs   17
  extended permissions   17, 18
acronyms   325
adjfs program   148
AIX V4.1
  See also packaging, AIX V4.1
  backup procedure   59
  compatibility filesets   295
  device driver packages   319
  device reconfiguration shell scripts   317
  exploiting features of   60
  file systems, compatibility   66
  fileset mapping from AIX V3.2   311
  levels for SMPs   174
  LVM and file systems features   65
  mksysb enhancements   57
  mksysb limitations   59
  packaging   238, 283
  platform types   175
  prototype files   175
  rmlpp script   307
  server vs. client for V4.1.3   67
  server vs. client for V4.1.4   68
  software maintenance   289
  system backup   57
  upgrading AIX client to AIX server   66
  VRMF numbering   289
All About AIX Version 4.1   xxi
alog command   44, 117
alternate cell   75
Animate command   101, 113
assist_migrate script   151
asynchronous I/O   230, 271
at command   14
authentication method   28
auto_merge, configuration file keyword   145
automatic database startup   269
automsg script   133

B

backing up a database   238, 253, 260
backing up AIX V3.2.5   30, 238
  ACLs   32
  backup procedure   32
  raw devices   32
backing up AIX V4.1   57, 59
berror program   97
BI_Error function   97
bi_main script   95
  See also installation process
  debugging exits   98, 101, 115
  error handling   95
    berror   97
    BI_Error   97
    ERROR_EXIT   97
    ERRORNUM   97
    ERRORTEXT   97
    loopled   95
  tracing   115
bi.log file   117
bidata program   101, 108
boot phase one   93
boot phase two   95
boot phases   92
boot process   91
  boot file system   91
  boot phase one   93
  boot phase two   95
  boot record   91
  compressed boot file system   91
  compressed UNIX kernel   91
  decompression code   91
  init   92
  network boot   93
  phases   92
  RAM file system   91
  UNIX kernel   91
boot record   91
bootinfo command   175
bootlist command   141
bos.rte.*.rmlist files   120, 145, 146
bos.rte.cfgfiles file   120, 145, 151
  See also configuration files
  keywords   145
bos.rte.post_i script   128, 145
bos.rte.pre_i script   120, 145, 146
bos.rte.root.rmlist file   120, 145, 146
bos.rte.usr.rmlist file   120, 145, 146
bosboot command   141, 177

ITSO Technical Bulletin Evaluation
RED000

International Technical Support Organization
A Holistic Approach to AIX V4.1 Migration, Volume 1
AIX, UP to SMP and Oracle
February 1996

Publication No. SG24-4652-00

Your feedback is very important to help us maintain the quality of ITSO Bulletins. **Please fill out this questionnaire and return it using one of the following methods:**

- Mail it to the address on the back (postage paid in U.S. only)
- Give it to an IBM marketing representative for mailing
- Fax it to: Your International Access Code + 1 914 432 8246
- Send a note to REDBOOK@VNET.IBM.COM

Please rate on a scale of 1 to 5 the subjects below.
(1 = very good, 2 = good, 3 = average, 4 = poor, 5 = very poor)

Overall Satisfaction   ____

Organization of the book	____	Grammar/punctuation/spelling	____
Accuracy of the information	____	Ease of reading and understanding	____
Relevance of the information	____	Ease of finding information	____
Completeness of the information	____	Level of technical detail	____
Value of illustrations	____	Print quality	____

Please answer the following questions:

a)   If you are an employee of IBM or its subsidiaries:

   Do you provide billable services for 20% or more of your time?   Yes____   No____

   Are you in a Services Organization?   Yes____   No____

b)   Are you working in the USA?   Yes____   No____

c)   Was the Bulletin published in time for your needs?   Yes____   No____

d)   Did this Bulletin meet your needs?   Yes____   No____

   If no, please explain:

   _____
   _____

What other topics would you like to see in this Bulletin?

   _____
   _____

What other Technical Bulletins would you like to see published?

   _____

Comments/Suggestions: (THANK YOU FOR YOUR FEEDBACK!)

_____   _____
Name   Address

_____   _____
Company or Organization

_____   _____
Phone No.

IBM®

Fold and Tape　　　　　　**Please do not staple**　　　　　　Fold and Tape
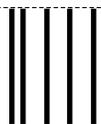
NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES
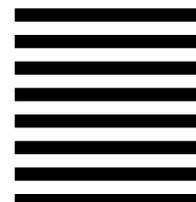
BUSINESS REPLY MAIL

FIRST-CLASS MAIL　　PERMIT NO. 40　　ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

IBM International Technical Support Organization
Department JN9B, Building 045
Internal Zip 2834
11400 BURNET ROAD
AUSTIN　TX
USA　78758-3493

Fold and Tape　　　　　　**Please do not staple**　　　　　　Fold and Tape

SG24-4652-00

IBM®

Printed in U.S.A.